Broadview®
www.broadview.com.cn

智能并非从人脑或机器中凭空产生
大数据是习得智能的必由之路

清华大学数据科学研究院
Tsinghua Institute for Data Science

清华大学数据科学研究院
清华大数据产业联合会 **联合力荐**

互联网时代的机器学习和自然语言处理技术
MACHINE LEARNING & NATURAL LANGUAGE PROCESSING IN THE INTERNET AGE

大数据智能

BIG DATA INTELLIGENCE

刘知远 崔安颀 等著

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

作者简介

刘知远，2011年清华大学博士毕业，现任清华大学计算机系助理研究员。研究兴趣为自然语言处理与社会计算。曾获清华大学优秀博士论文、中国人工智能学会优秀博士论文、清华大学优秀博士后等称号。
liuzy@tsinghua.edu.cn

崔安颀，2013年清华大学博士毕业，现任加拿大滑铁卢大学博士后研究员。参与智能问答创业，部分产品已在微信、手机等多种平台上线。研究兴趣为情感分析、问答系统与社交媒体分析。
caq@caq9.info

赵鑫，2014年北京大学博士毕业，现任中国人民大学信息学院计算机系教师。研究兴趣为社交媒体数据挖掘与自然语言处理。曾获北京大学优秀博士论文、微软学者等称号。
batmanfly@gmail.com

张开旭，2012年清华大学博士毕业，曾经和现在就职于BAT和创业公司。研究兴趣为自然语言处理与机器学习。
zhangkaixu@outlook.com

韩文弢，2015年清华大学博士毕业，现任清华大学计算机系博士后研究员。研究兴趣为计算机系统。曾带领学生团队搭建清华大学“学堂在线”MOOC平台初版并成功上线。
hanwentao@tsinghua.edu.cn

张永锋，清华大学计算机系博士生，加州大学圣克鲁兹分校访问学者。研究兴趣为信息检索、个性化推荐与计算经济学。曾获西贝尔学者、百度学者、微软学者等称号。
yongfengz@foxmail.com

互联网时代的机器学习和自然语言处理技术

MACHINE LEARNING & NATURAL LANGUAGE PROCESSING IN THE INTERNET AGE

大数据智能

BIG DATA INTELLIGENCE

刘知远 崔安颀 等著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书是一本介绍大数据智能分析的科普书籍，旨在让更多的人了解和学习互联网时代的机器学习和自然语言处理技术，以期让大数据技术更好地为我们的生产和生活服务。

全书包括大数据智能基础和大数据智能应用两个部分，共 8 章。大数据智能基础部分有三章：第 1 章以深度学习为例介绍大数据智能的计算框架；第 2 章以知识图谱为例介绍大数据智能的知识库；第 3 章介绍大数据的计算处理系统。大数据智能应用部分有 5 章：第 4 章介绍智能问答，第 5 章介绍主题模型，第 6 章介绍个性化推荐，第 7 章介绍情感分析与意见挖掘，第 8 章介绍面向社会媒体内容的分析与应用。最后在本书的后记部分为读者追踪大数据智能的最新学术材料提供了建议。

本书适合作为高等院校计算机相关专业的研究生学习参考资料，也适合电脑爱好者阅读。作者特别希望本书能够帮助所有愿意对大数据技术有所了解，以及想要将大数据技术应用于本职工作的读者。

图书在版编目（CIP）数据

大数据智能：互联网时代的机器学习和自然语言处理技术 / 刘知远等著. —北京：电子工业出版社，2016.1
ISBN 978-7-121-27648-4

Ⅰ. ①大… Ⅱ. ①刘… ②崔… Ⅲ. ①机器学习②自然语言处理 Ⅳ. ①TP181②TP391

中国版本图书馆 CIP 数据核字（2015）第 281768 号

统筹策划：顾慧芳
责任编辑：徐津平
特约编辑：顾慧芳
印　　刷：三河市双峰印刷装订有限公司
装　　订：三河市双峰印刷装订有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编　100036
开　　本：787×980　1/16　印张：14.75　字数：322 千字
版　　次：2016 年 1 月第 1 版
印　　次：2016 年 5 月第 2 次印刷
印　　数：3001～5000 册　　定价：49.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

前 言

天才并不是自生自长在深林荒野里的怪物，是由可以使天才生长的民众产生、长育出来的，所以没有这种民众，就没有天才。

——鲁迅

千淘万漉虽辛苦，吹尽狂沙始到金。

——[唐]刘禹锡

大数据时代与人工智能

在进入21世纪前后，很多人预测这将会是怎样的世纪。有人说这将是生命科学的时代，也有人说这将是知识经济的时代，不一而足。现在15年过去了，随着互联网的高速发展，大量的事实强有力地告诉我们，这必将是大数据的时代，是智能信息处理的黄金时代。

自2012年美国奥巴马政府发布大数据研发倡议以来，关于大数据的研究与思考在全球蔚然成风，已经有很多专著面世，既有侧重趋势分析的，如舍恩伯格和库克耶的《大数据时代》（盛杨燕和周涛教授译），涂子沛的《大数据》和《数据之巅》，也有偏重技术讲解的，如莱斯科夫等人的《大数据》（王斌教授译）、张俊林的《大数据日知录》、杨巨龙的《大数据技术全解》，等等。相信随着大数据革命的不断深入推进，会有更多的专著出版。

前人已对大数据的内涵进行过很多探讨与总结，其中比较著名的是所谓的3V定义：大容量（Volume）、高速度（Velocity）和多形态（Variety）。3V的概念最早于2001年由麦塔集团（Meta Group）分析员道格·莱尼（Doug Laney）提出，后来被高德纳咨询公司（Gartner）正式用来描述大数据。此外还有很多研究者提出更多的V来描述大数据，例如真实性（Veracity），等等。既然有如此众多珠玉在前，我们推出这本书，当然希望讲一些不同的东西，这点不同的东西就是智能。

人工智能一直是研究者们非常感兴趣的话题，并且由于众多科幻电影或小说作品的影响而广为人知。1946 年第一台电子计算机问世之后不久，英国著名学者图灵就发表了一篇重要论文（题名《计算机器与智能》*Computing Machinery and Intelligence*），探讨了创造具有智能的机器的可能性，并提出了著名的“图灵测试”，即如果一台机器与人类进行对话，能够不被分辨出其机器的身份，那么就可以认为这台机器具有了智能。自 1956 年达特茅斯研讨会正式提出了“人工智能”的研究提案以来，人们开始了至今长达半个多世纪的曲折探索。

我们且不去纠结“什么是智能”这样哲学层面的命题（有兴趣的读者可以参阅罗素和诺维格的《人工智能——一种现代方法》*Artificial Intelligence: A Modern Approach* 以及杰夫·霍金斯的《智能时代》*On Intelligence*），而是先来谈谈人工智能与大数据有什么关系？要回答这个问题，我们来看一个人是如何获得智能的。一个呱呱坠地、只会哭泣的婴儿，最后长成思维健全的成人，至少要经历十几年与周围世界交互和学习的过程。从降临到这个世界的那一刻起，婴儿无时无刻不在通过眼睛、耳朵、鼻子、皮肤接收着这个世界的信息：图像、声音、味觉、触觉，等等。你有没有发现，无论从数据的规模、速度还是形态来看，这些信息无疑是典型的大数据。因此，人类习得语言、思维等智能的过程，就是从大数据学习的过程。智能不是无源之水，它并不是凭空从人脑中生长出来的。同样，人工智能希望让机器拥有智能，也需要以大数据作为学习的素材。可以说，大数据将是实现人工智能的重要支撑，而人工智能是大数据研究的重要目标之一。

但是，在人工智能研究早期人们并不这样认为。早在 1957 年，由于人工智能系统在简单实例上的优越性能，研究者们曾信心满怀地认为，10 年内计算机将能成为国际象棋冠军，而通过简单的句法规则变换和词典单词替换就可以实现机器翻译。事实证明，人们远远低估了人类智能的复杂性。即使在国际象棋这样规则和目标极为简单清晰的任务上，直到 40 年后的 1997 年，由 IBM 推出的深蓝超级计算机才宣告打败人类世界冠军卡斯帕罗夫。而在机器翻译这样更加复杂的任务（人们甚至连优质翻译的标准都无法达成共识，并清晰地告诉机器）上，计算机至今还无法与人类翻译的水平相提并论。

当时的问题在于，人们远远低估了智能的深度和复杂度。智能是分不同层次的。对于简单的智能任务（如对有限句式的翻译等），我们当然可以简单制定几条规则就能完成。但是对于语言理解、逻辑推理等高级智能，简单方法就束手无策了。

生物界从简单的单细胞生物进化到人类的过程，也是智能不断进化的过程。最简单的单细胞生物草履虫，虽然没有神经系统，却已经能够根据外界信号和刺激进行反应，实现趋利避害，我们可以将其视作最简单的智能。而巴甫洛夫关于的狗的条件反射实验，则向我们证明了相对更高级的智能水平，可以根据两种外界信号（铃声与食物）的关联关系，

实现简单的因果推理，也就是能根据铃声推断食物即将出现。人类智能则是智能的最高级形式，拥有了语言理解、逻辑推理与想象等独特的能力。我们可以发现，低级智能只需小规模的简单数据或规则的支持，而高级智能则需要大规模的复杂数据的支持。

同样重要的，高级智能还需要独特计算架构的支持。很显然，人脑结构就与狗等动物有着本质的不同，因此，即使将一只狗像婴儿一样抚育，也不能指望它能完全学会理解人类的语言，并像人一样思维。受到生物智能的启示，我们可以总结出如图 0.1 所示的基本结论，不同大小数据的处理，需要不同的计算框架，带来不同级别的智能。

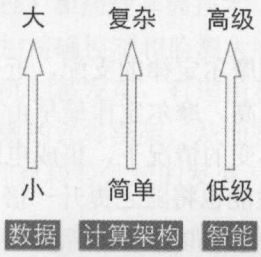


图 0.1 不同规模的数据需要不同的计算架构，产生不同级别的智能

人工智能是否要完全照搬人类智能的工作原理，目前仍然争论不休。有人举例，虽然人们受到飞鸟的启发发明了飞机，但其飞行原理（空气动力学）却与飞鸟有本质不同；同样，生物界都在用双脚或四腿奔跑行走，人们却发明了轮子和汽车实现快速移动。然而不可否认，大自然无疑是我们最好的老师。人工智能固然不必完全复制人类智能，但是知己知彼，方能百战不殆。生物智能带来的启示已经在信息处理技术发展中得到了印证。谷歌研究员、美国工程院院士 Jeff Dean 曾对大数据作过类似结论：“对处理数据规模 X 的合理设计可能在 10X 或 100X 规模下就会变得不合理”（Right design at X may be very wrong at 10X or 100X.），也就是说，大数据处理也需要专门设计新颖的计算架构。而与人工智能密切相关的机器学习、自然语言处理、图像处理、语音处理等领域，近年来都在大规模数据的支持下取得了惊人进展。我们可以确信地说，大数据是人工智能发展的必由之路。

大数据智能如何成真

虽然大数据是实现人工智能的重要支持，但如何实现大数据智能，却并非显而易见。近年来随着计算机硬件、大数据处理技术和深度学习等领域取得了突破性进展，涌现出一批在技术上和商业上影响巨大的智能应用，这让人工智能发展道路日益清晰起来。

大数据的价值并非水落石出这样显而易见。我们认为，近年来人工智能的突破性进展，主要是在触手可及的人类社会大数据、高性能的计算能力以及合理的智能计算框架的支持下，方能披沙拣金实现大数据智能。

人类社会大数据触手可及。如前所述，这是大数据的时代，互联网的兴起，手机等便携设备的普及，让人类社会行为数据越来越多地汇聚到网上，触手可及。这让机器从这些大数据中自动学习成为可能。但是，大数据（如大气数据、地震数据等）并非现在才出现，只是在过去我们限于计算能力和计算框架，难以从中萃取精华。因此，大数据智能的实现还依赖以下两个方面的发展。

（1）计算能力突飞猛进。受到摩尔定律的支配，近半个世纪以来，计算机的计算和存储能力一直在以令人目眩的速度提高。摩尔定律最早由英特尔（Intel）创始人之一戈登·摩尔提出，基本思想是：保持价格不变的情况下，集成电路上可容纳的元器件的数目大约每隔 18 到 24 个月就会增加一倍，性能也将随之提升一倍。也就是说，每一块钱能买到的计算机性能将每隔 18 到 24 个月提升一倍以上。虽然人们一直担心，随着微处理器器件尺寸变小，摩尔定律会受到量子效应影响而失效。但至少从已有发展历程来看，随着多核、多机并行等新框架的提出，计算机已经能够较好地支持大规模数据处理所需的计算能力。

（2）计算架构返璞归真。近年来，深度学习在图像、语音和自然语言处理领域掀起了一场革命，在图像分类、语音识别等重要任务上取得了惊人的性能突破，在国际上催生了苹果 Siri 等语音助手的出现，在国内则涌现了科大讯飞、Face++等高科技公司。然而我们可能很难想象，深度学习的基础“人工神经网络”技术，此前曾长期处于无人问津的境地。在深度学习兴起以前，人工神经网络常被人诟病存在可解释性差、学习稳定性差、难以找到最优解等问题。然而，正是由于大规模数据和高性能计算能力的支持，才让以人工神经网络为代表的机器学习技术在大数据时代焕发出蓬勃生机。

人工智能的下一个里程碑

当下，以深度学习为代表的计算框架在很多具体任务上取得了巨大成绩。甚至有媒体和公众已经开始恐慌人工智能取代人类的可能性。然而，从理性来看，深度学习的处理能力和效率与人类大脑相比仍有巨大差距。因此，仅依靠大数据智能，并非孕育人工智能的终极之道。随着技术的进步和研究的深入，现有解决方案必然触及无法逾越的天花板，进入瓶颈期。

人脑拥有现有计算框架不可比拟的优势。例如，虽然人脑中信号传输速度要远低于计算机中的信号速度，但是人脑在很多智能任务上的处理效率则远高于计算机，例如在众多声音中快速识别出叫自己名字的声音，通过线条漫画认出名人，复杂数学问题的推导求解，快速阅读理解一篇文章，等等。可以想象，在计算速度受限的条件下，人脑一定拥有某种独特的计算框架，才能完成这些叹为观止的智能任务，可谓大自然的鬼斧神工。

那么人工智能的下一个里程碑是什么呢？我猜想可能是神经科学及其相关学科。一直以来，神经科学都在探索各种观测大脑活动的工具和方法，并作出了大量的实证和建模工作。随着光控基因技术（optogenetics）和药理基因技术（pharmacogenetics）等新技术的发展，人们拥有了在时间和空间上更加精确控制和监测大脑活动的能力，从而有望彻底发现人脑的神经机制。一旦人脑的神经机制被发现，有理由相信，人们可以迅速通过仿真等方式，在计算机中实现类似甚至更高效的计算框架，从而推动实现人工智能的最终目标。此外，量子计算、生物计算、新型芯片材料等领域的发展，都为我们展现出无限可能的未来。

当下，社会大数据、计算能力和计算框架三方面的发展融合产生出了大数据智能。我们相信更大规模数据、更强计算能力和更合理计算框架的推出，会不断推动人工智能向前发展。然而，正如前几年社会各界对物联网、云计算的追捧，最近社会对大数据和人工智能概念的炒作愈演愈烈，产生很多不切实际的幻想和泡沫。对于这个领域重新得到青睐，我们当然感到欣慰。但是，也不妨多一些谨慎和冷静。鉴古知今，回顾人工智能的曲折发展史（《人工智能——一种现代方法》中有详细介绍）我们看到，在过度的期望破灭之后，随之而来的往往就是严冬。现在大数据智能万众瞩目，我们不妨心中默念凛冬将至。

事物总是在不断自我否定中螺旋前进的，人工智能的探求之路也是如此。我们相信大数据是获得智能的必由之路，但现在的做法不见得就一定正确。多年之后，我们也许会用截然不同的办法处理大数据。然而这些都不重要，重要的在于一颗无论冷门热门都执着的心、坚持不懈的信念。就像现在深度学习领域的巨人 Geoffrey Hinton、Yann LeCun 等学者，在这之前坐了十几年的冷板凳，研究成果屡屡被拒。对于真正的学者，研究领域冷门热门也许都不重要，反而会成为对从业者的试金石——只有在寒冬中坚持下来的种子，才能等到春天绽放。

关于本书

这本书并不想在已经火得发紫的大数据火堆上再添一把柴。这本书希望从人工智能这个新的角度，总结大数据智能取得的成果，它的局限性以及未来可能的发展前景。

本书从大数据智能基础和应用两个方面展开介绍。

基础部分有三章：第 1 章以深度学习为例介绍大数据智能的计算框架，第 2 章以知识图谱为例介绍大数据智能的知识库；第 3 章介绍大数据的计算处理系统。

在大数据智能的应用部分，我们选择文本大数据作为主要场景进行介绍，主要原因在于，语言是人类智能的集中体现，语言理解也是人工智能的终极目标，图灵测试的设置是以语言作为媒介的。应用部分有五章：第 4 章介绍智能问答，第 5 章介绍主题模型，第 6 章介绍个性化与推荐，第 7 章介绍情感分析与意见挖掘，第 8 章介绍面向社会媒体内容的分析应用。这基本涵盖了文本大数据智能处理的主要应用场景。以后如有机会再版，还计划纳入文档摘要、计算广告学等主题。

大数据智能仍然是个高速发展的领域。可以想象这本书出版的时候，很多内容已显陈旧。为了让读者能够跟踪这个领域的最前沿进展，本书专门设置后记，为初学者追踪大数据智能的最新学术材料提供建议。

个人学识有限，深怕在自己不擅长的领域说出外行话甚至错误连篇。因此，我邀请熟识的同学朋友撰写他们所擅长的章节。除了前言、第 2 章、第 8 章和后记由我操刀外，我请同门师弟张开旭博士撰写第 1 章，清华大学计算机系统方向博士韩文弢撰写第 3 章，清华大学信息检索方向博士崔安颀撰写第 4、7 章，北京大学自然语言处理方向博士、现中国人民大学信息学院教师赵鑫撰写第 5 章，清华大学个性化推荐方向博士生张永锋同学撰写第 6 章。他们都在相关领域开展了多年研究工作，发表过高水平论文。最后，我对全书做了统稿和校对，北京邮电大学毕业的林颖同学在我们实验室实习期间帮助我做了大量的书稿整理工作。

致谢

本书能够出版，无疑得到了很多人的支持和帮助。

首先，感谢这本书的几位合作者张永锋、崔安颀、张开旭、赵鑫和韩文弢，他们的热情、无私与认真，让我相信这本书能够真的为读者提供及时有用的知识。

其次，感谢我的导师和领导清华计算机系的孙茂松教授，是他将我带入了这个精彩纷呈的研究领域，也是他为我提供了宽松的写作环境，能够让这本书顺利问世。

我还要感谢刘洋（清华大学）、付杰（新加坡国立大学）、来思惟（中科院自动化所）

等同事、同学和好友，在本书撰写过程中提供了很多最新进展和热情帮助。特别感谢林颖同学所做的书稿整理和封面设计工作。

最后，我要特别感谢电子工业出版社副总编辑兼计算机分社社长郭立老师的热情邀请和大力支持，以及本书编辑、清华计算机系学长顾慧芳老师的不断激励和鼎力相助，让我鼓起勇气敢于接下这个选题，也能在我拖延症反复发作时耐心地等待，经过了两年多时间的酝酿、收集资料、研究分析以及整理撰写，终于变成了你手中的这本书。

欢迎交流

当今世界，大数据智能是一个涉及非常广泛、而且发展非常迅猛的领域，这个领域的研究成果将帮助人类加速认识世界、探索宇宙，也将极大地影响到人们日常生活的方方面面。因此，笔者想在从事学习和自然语言处理等基础技术和最新进展研究工作的同时撰写一本介绍这一领域的科普书籍，作为抛砖引玉，旨在为需要了解与学习大数据智能技术的朋友提供帮助，甚至加入到大数据智能分析这一充满惊奇和魅力的领域中来。

当然，笔者尽量以开放的态度梳理每个方向的相关成果和进展，然而大数据智能日新月异，而我们所知有限，难免有挂一漏万之憾。如有重要进展或成果没有被介绍到，绝非作者故意为之，敬请大家批评指正。我们欢迎读者对本书的任何反馈，无论是指出错误还是改进建议，请直接发邮件给我：liuzy@tsinghua.edu.cn。我们会专门开辟网站维护勘误清单，如果本书有机会再出下一版的话，也会尽量改正所有发现的错误。

刘知远博士
清华大学计算机科学与技术系　助理研究员
2015 年 8 月于北京清华园

目 录

第1章

深度学习——机器大脑的结构

为了实现高层抽象表征的复杂能力，我们需要深层结构。

——[美]尤舒·本吉奥（Yoshua Bengio）

1.1 概述

深度学习（Deep Learning）是近年来兴起的机器学习范式，已经被谷歌、微软、百度等互联网巨头投入巨资进行相关研发，甚至提出“谷歌大脑”等大型项目计划。深度学习利用多层神经网络结构，从大数据中学习现实世界中各类事物能直接被用于计算机计算的表示形式（如图像中的事物、音频中的声音等），被认为是智能机器可能的“大脑结构”。

虽然没有像云计算、物联网等概念一样在普通大众中被炒得沸沸扬扬，但深度学习真的火了一把。不论在学术界还是在工业界，深度学习刮起了一阵不小的旋风。在语音识别、图像识别、自然语言处理等研究领域，掀起了一次深度学习的热潮。在某些老问题上，它摧枯拉朽，颠覆了使用多年的老方法；在另一些前沿问题中，它完全不同于先前的流行方法，却又实现了惊人的效果提升；更为重要的，它引起了对主流方法的反思，提供了一个全新的视角，在今后势必会催生出更多的研究成果。

深度学习的实用性也被工业界发现，它不但能够取代一些传统方法，还使得某些停留在概念阶段的应用更接近实用。本章将与读者一起探讨以下问题：深度学习是什么，它的哪些特点让人眼前一亮，它离终极神器还有多远，它给了我们哪些有意义的思考。

本章默认读者拥有最基本的计算机科学知识和微积分、线性代数和概率论知识，但不需要具有人工智能方面的知识。了解深度学习并不需要很有“深度”的知识，简单地说深度学习就是：使用多层神经网络来进行机器学习。

为了在一节中将深度学习的基本面貌展现出来，首先我们需要了解神经网络，它是一个带参数的函数，通过调整参数，可以拟合不同的函数。而机器学习就是一种让计算机自动调整函数参数以拟合想要的函数的过程。以上即为“Deep Learning”的“Learning”部分。

接着我们将介绍，多个这种带参数的函数可以进行嵌套，构成一个多层神经网络，能更好地拟合某些实际问题中需要的函数。但是可惜的是这个方法提出了几十年，学者们却没有找到在这种情况下有效地自动调整函数参数的算法。直到前些年逐层预训练的方法被提出，才使得这一方法能够达到较好的效果。以上即为“Deep Learning”的“Deep”部分。

1.1.1 可以做酸奶的面包机——通用机器的概念

最近家里买了一台面包机，我觉得这个小机器非常有意思。在我看来，它在形式上非常

像一个全自动洗衣机——只需要在开始的时候往一个缸体内加入原材料面粉、水（脏衣服）以及一些辅料如酵母、盐（洗衣液），然后选择模式如普通面包、法式面包等（快洗、大件衣服等），盖上盖子，按“开始”按钮。机器就开始按照编好的程序进行搅拌、加热等操作（洗衣机还会进水、放水）。最后在提示音之后打开盖子，就得到了我们想要的输出——面包（干净衣服）。当然，还可以根据需要只进行某部分操作，如只进行和面或发酵（只漂洗或甩干）。

而面包机有意思的特别之处是，它还可以用来做酸奶和醪糟（一种发酵的米酒），这是因为加热发酵过程对于制作面包、酸奶和醪糟是类似的。也就是说，一个可控温度的容器是一个通用的工具，给它不同的运行参数（时间、温度），它就可以实现不同的功能。在我读大学的时候，也听说生物系的同学用一种叫做“电热恒温培养箱”的实验仪器自制酸奶，这种机器做酸奶跟面包机相比就如同单反相机跟数码相机的区别吧，如图 1.1 所示。

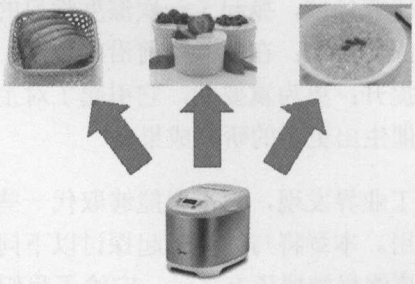


图 1.1 可以制作面包、酸奶和醪糟的面包机

通过以上的叙述，我想说的是一个工具或者机器的参数，通常包含固定的和可调的两部分。一个通用性强的机器，调整其可调的参数，就可以实现不同的功能。这适用于面包机、洗衣机、电热恒温培养箱、照相机，等等。而我们的计算机其实也是一样的，它也可以看作一种参数高度可调的工具，所谓计算机的参数，就是某个程序的代码，运行不同的代码就可以实现不同的功能。

而本章将要介绍的深度学习所基于的人工神经网络[1]（在之后若不加说明，神经网络均是指人工神经网络），也是一种参数高度可调的通用工具。它的本质是一个由一系列向量、矩阵运算构成的函数，函数的输入和输出就是这个工具的输入和输出，函数表达式中某些矩阵和向量的取值，就构成了这个工具可调的参数，而函数表达式形式本身，则是不可调的。

例如，函数 $y=f(x; a, b)=ax+b$ 就可以被看作一个人工神经网络。其输入为 x 输出为 y，可调整的参数为 a 和 b。不同的参数，可以构成不同的函数以实现不同的功能。

1 人工神经网络是 20 世纪 80 年代以来人工智能领域兴起的一个研究热点。它从信息处理角度对人脑神经元网络进行抽象，建立某种简单模型。

1.1.2 连接主义

我们再接着讨论一下神经网络这种通用机器的特点。

神经网络的理念是，给出一种通用的函数形式，它的计算步骤、方式均是固定的，其功能只由其中的参数取值决定。也就是说，其参数是一些实数向量和矩阵。所有参数构成的参数空间是一个无约束的高维欧氏空间。空间中任意一个点就是一组具体的参数取值，也就对应于一种实现具体功能的工具。

说得形而上一些，这是一种叫做连接主义的理念。可以与我们的人脑类比，连接主义认为人脑就是一种这样的通用机器，构成人脑的脑细胞所实现的功能很固定很简单，人之所以拥有智能，则是因为数量庞大的这样的简单单元以某种非常特殊的方式互相连接着。在之后的小节中我们会看到，人工神经网络可以看作是对人脑某种程度的模拟，人工神经网络中的函数相当于定义了某种特殊的脑细胞的连接方式，而函数中可调的参数则定义了在这种连接方式下这些连接的强度。

我们还能举出一些与连接主义不同的理念。

例如，我的中学化学老师一直挂在嘴边的一句话就是“结构决定功能”。化合物的功能，是由其结构决定的。当我们写出 H_2O 时，它的化学性质已经确定了，诸如氧原子和氢原子的连接强度这样的信息是次要的。这与人工神经网络中连接权重决定功能是不一样的。

在人工智能领域，与连接主义对立的有“符号主义”。后者并不是将人的智能解释为脑细胞的某种连接方式，而是将其解释为人类与其他动物相比非常强的符号处理能力。例如人类的语言，就可以看作一个符号系统。符号系统的特点是其对象是离散的，例如细致分析每个人发“machine”的声音可能千差万别，但只要与其他单词能够区别，我们就把它当作同一个符号，而这个符号又对应着某种意义，比如“machine”的意义就是“能完成某种功能的工具或装置”。更为复杂的，对符号的处理还包括对符号串的处理以及对符号逻辑的处理。在我们已知的生物中，只有人类才掌握了如此强的符号处理能力。

那么，处理符号的函数与定义人工神经网络的函数的区别就很明显了。用来处理符号的函数的输入应该是符号性的，即离散的变量及其组合，不同的符号和它们的组合都有其背后对应的意义，而人工神经网络的输入通常是实数构成的向量或矩阵，每个分量不需要有抽象的意义。处理符号的函数会用到很多逻辑运算，而人工神经网络一般使用代数运算。处理符号的函数中可调整的参数会比较抽象，参数空间由符号的组合来表示（类似于我们中学学的排列组合），而人工神经网络中可调的参数也是一些实数构成的向量和矩阵，参数空间是欧氏空间。

1.1.3 用机器设计机器

我们现在使用的所有计算机都等价于图灵机[1]，Google 为纪念图灵推出的“图灵机”Doodle 如图 1.2 所示，程序员设计出程序（机器的参数）并运行机器，计算机就可以完成不同的任务。而人工神经网络的野心并不是提出另一个计算框架，让人手工地在这个框架下调整神经网络的参数，然后构造不同功能的机器。它是想给出一种通用的算法，能够自动地找到一组参数的取值，让神经网络能够很好地完成给定任务。

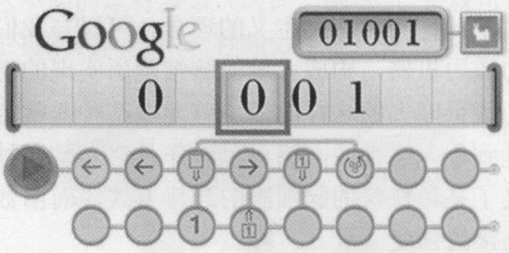


图 1.2 Google 为纪念图灵推出的“图灵机”Doodle

一言以蔽之，人工神经网络的目标是用机器设计机器。

这可是个能够让人心跳加速的口号。人类的祖先花了成百上千万年的进化，学会了制造工具，进入了石器时代。又花了几百万年的时间，学会了用机器制造机器，发生了工业革命。下一个里程碑就应该是用机器设计机器，注意这里不是指用机器辅助人类设计机器，如现在已经有的计算机辅助设计（CAD），而是指机器根据人的需求自己设计机器。

机器设计机器，在人工神经网络中，就是机器自动调整参数。更具体地，就是在一个高维欧氏参数空间中，找到一点，对应于一种机器，能够完成某种指定的任务。

在后面的章节中会介绍在人工神经网络中，是通过怎样的方式以及算法使得一个神经网络可以根据不同的问题自动地调整参数。

1.1.4 深度网络

到目前为止，我们只介绍了一般的人工神经网络模型。对于深度学习，其使用的人工

1 图灵机又称图灵计算机，是由数学家阿兰·麦席森·图灵（1912～1954）提出的一种抽象计算模型，即将人们使用纸笔进行数学运算的过程进行抽象，由一个虚拟的机器替代人们进行数学运算。

神经网络是一种多层前馈神经网络。用数学的语言描述，就是说神经网络所对应的函数是一个由多个函数嵌套而成的函数。例如：

$$f(x)=g_3(g_2(g_1(x;w_1);w_2);w_3)$$

函数$f(x)$由三个函数嵌套而成。三个函数各有自己的参数。

不过，以上所描述的仍旧是一个也许比大多数读者都要年长的模型。为何在近年它才再次火起来呢，因为虽然该模型被提出很久，但是学者们一直无法找到很好的方法用以调整各个层次函数中的参数。也就是说，只是由于解决了一个纯技术性的问题，使得这个老模型再次成为人们的焦点。

一个由嵌套函数构成的函数是非常复杂的，参数空间非常大且复杂，会很容易只能找到局部最优的解，而这样的解往往并不太好，从而使得这种模型的效果反而不如其他简单模型。近年，学者终于找到了第一种能够有效训练参数的方法，即逐层预训练并最后进行微调。所谓逐层预训练，就是用一种方法逐个训练。例如对于式$f(x)=g_3(g_2(g_1(x;w_1);w_2);w_3)$代表的神经网络，就是用不同的方法逐个地分别确定参数w_1、w_2、w_3的值。最后再合在一起进行微调。而如何逐个学习各层的参数，其中的妙处我们将在后面的章节进行介绍。

1.1.5 深度学习的用武之地

以上讨论了何为机器学习，以及神经网络这种特殊的实现机器学习目标的框架，而深度学习就是指一类特殊的神经网络。

深度学习作为一种机器学习模型，近年来在很多应用场合都取得了很好的成绩。以下按机器学习任务的应用类别来加以分别介绍。

（1）**分类**。分类任务的输出为一个离散的量。例如在垃圾邮件自动过滤中，输入一封邮件，输出是或不是垃圾邮件。图像识别是一种具体的分类任务，其输入是一幅图片，输出为图片中物体的名字（一般是在一个有限的集合中找出一个）。Andrew Ng 是这方面的牛人。他与 Google 建立的团队使用 1000 台机器，16000 个 CPU 组成集群构成一个神经网络，对 20000 种物体的图片进行识别。相对于非深度学习的方法的准确率提高了 70%。最近，Ng 教授的团队继续用 GPU 取代 CPU 进行计算，使得可以用少得多的机器完成相同任务（Raina, et al. 2009）。

（2）**结构分类**。结构分类是一类特殊的分类问题，其输出不是一个简单的离散的量，而是由多个离散的量构成的结构。例如我们文字交流中使用的句子，即是由离散的量（汉

字或单词）线性连接构成的。语音识别就是一个非常有用的结构分类问题，其输入是一段语言的录音，输出是语音对应的句子。供职于微软的邓力使用深度学习在语音识别方面取得了很大的进步。在一个大会上，微软演示了一个可以进行实时翻译的系统，演讲者用英文演讲，其语言被实时识别为英文文本，再被翻译为汉语，最后合成相应的汉语语音被播放出来，技惊四座。句法分析（Socher, et al. 2013）、情感分析（Socher, et al. 2012）、机器翻译（Devlin, et al. 2014）都是深度学习在结构分类中应用的典型场景。

（3）**回归**。回归是不同于分类的另一大类问题。它的输出不是离散的量，而是连续的实数。语言模型就是一个特殊的回归任务，对于一个句子，它给出这个句子产生的概率。常用句子的概率高于不常用的句子，正确的句子的概率高于错误的（例如不合语法的）句子。深度学习的鼻祖之一 Hinton 就曾用深度网络构造过高质量的语言模型（Mnih & Hinton 2009）。语言模型在语音识别、机器翻译、中文输入法中都起到很大的作用。

1.2 从人脑神经元到人工神经元

本部分将介绍学者们是如何从人脑神经元中得到灵感，并设计出一种与人脑神经元类似但又有简洁数学表示的计算单元人工神经元。

1.2.1 生物神经元中的计算灵感

早在 1771 年，意大利的路易吉·伽伐尼就发现，如果用电火花刺激死青蛙的肌肉就能使其颤动（可怜的青蛙，不知道我们中学做的往被切除大脑的青蛙身上涂硫酸的实验是否也与这位科学家有关）。1848 年，德国人 Emil du Bois-Reymond 发现了神经元受到激发而产生的动作电位。1906 年意大利的卡米洛·高尔基和西班牙的圣地亚哥·拉蒙·卡哈尔由于神经系统的发现而共同获得了诺贝尔奖。

我们可以这样简单地描述一个生物神经元[1]的工作机制。一个神经元就是一个可以接收、发射脉冲信号的细胞。在细胞的核心之外有树突与轴突，树突接收其他神经元的脉冲信号，而轴突将神经元的输出脉冲传递给其他神经元，一个神经元传递给不同神经元的输出是相同的。一个神经元的状态有两种——非激活和激活。非激活的神经元不输出脉冲，

1 人的大脑大约由 140 亿个神经元组成，神经元连接成神经网络。神经元是大脑处理信息的基本单元，以细胞体为主体，由许多向周围延伸的不规则树枝状纤维构成的神经细胞，它是大脑处理信息的基本单元。

激活的神经元会输出脉冲。神经元激活与否由其接收的脉冲决定。

模拟人脑神经元的历史几乎跟现代计算机的历史一样久远。1943 年，生理学家 McCulloch 和数学家 Pitts 提出了第一个神经网络的数学模型，其中神经元的模型，仍然适用于近几年开始流行的深度学习。冯·诺意曼在 1955-1956 间写过一个演讲稿，后来被商务出版社以《计算机与人脑》为题出版，编入汉译世界学术名著丛书，与《理想国》、《新工具》等书并列。在那个数字计算机和模拟计算机并存的年代，作者在诸多方面对计算机和人脑进行了比较。

根据 McCulloch 和 Pitts 的模型，可以将一个神经元看作一个计算单元，对输入进行线性组合，然后通过一个非线性的激活函数作为输出。用 $x_1, x_2, \ldots x_n$ 表示 n 个输入，用 y 表示输出，用 f 表示非线性的激活函数，则有

$$y = f(u) = f(x_1 w_1 + \cdots x_n w_n + b)$$

其中 b 是一个与输入无关的量。将以上模型表示为图形的结构即如图 1.3 所示。

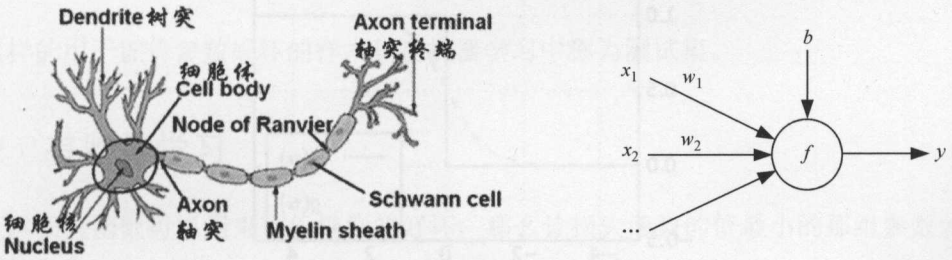


图 1.3 神经元与神经元模型

在这里，x 与 y 分别是输入与输出，它们可以是任意的实数。w 和 b 是模型的参数，不同的参数会构成实现不同功能的模型。

事实上，在真正的人脑神经元中，输出与输入有更复杂的关系。

1.2.2 激活函数

不同的激活函数适合不同的具体问题和神经网络参数学习算法。如果期望得到离散的输出，可以使用阶跃激活函数

$$s(u) = 1(u > 0)$$

$$s(u)=0\left(u\leqslant 0\right)$$

如果期望得到连续的输出，可以使用一类被称为 sigmoid 的 S 形函数。一种在应用中使用得比较多的是 Logistic 函数：

$$g(u)=\frac{1}{1+e^{-u}}$$

注意，如果希望值域是（−1, 1）而不是（0, 1），激活函数还可以定义为双曲正切函数，它跟 Logistic 函数非常类似：

$$\tanh(u)=\frac{1-e^{-2u}}{1+e^{-2u}}=2g(u)-1$$

阶跃函数与 logistic 函数的对比可见图 1.4。激活函数通常都是单调有界的。

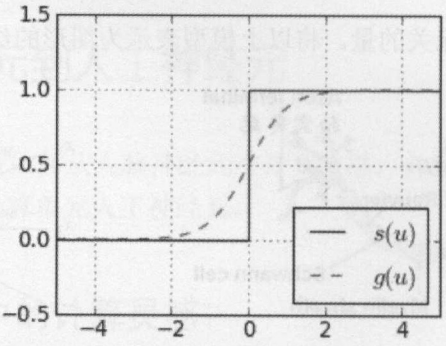


图 1.4 阶跃函数与 Logistic 函数对比

此外，Logistic 函数处处连续可导，并且导数均大于 0，这为之后的参数学习提供了方便。

1.3 参数学习

上一节我们介绍了人工神经网络如何脱胎于生物实体，成为一个纯粹的数学模型。这一节我们会看到这个数学模型又如何从一个理论模型，变为工程上可实际运用的工具。

这一节所考虑的问题是，在神经网络结构确定之后，如何确定其中的参数，即上一节中针对单个神经元的 $\boldsymbol{w}$ 以及 b。为了描述得简洁，在本节中使用 $\boldsymbol{w}$ 表示所有需要确定的参数。

1.3.1 模型的评价

首先我们要讨论如何评价模型参数的好坏，否则讨论“如何选择好的参数”这个问题就是没有意义的。这里介绍的是机器学习领域的一般方法。

一个好的神经网络，对于给定的输入，能够得到设计者期望的输出。设输入为 $\boldsymbol{x}$，输出为 $\boldsymbol{y}$，我们期望的输出为 $\boldsymbol{y}^*$，可定义实际输出与期望输出的差别作为评价神经网络好坏的指标。例如可以将两个向量的距离的平方定义为这个差别

$$d(y^*, y) = \frac{1}{2} | y^* - y |^2$$

只考察单个样本还不够充分，通常需要考察一个样本的集合，例如有 $(x_i, y_i^*) | i=1...N$，定义损失函数为：

$$\text{loss}(w) = \sum_{i=1}^{N} d(y_i^*, y_i)$$

这样的用于评价参数好坏的样本集在机器学习中称为测试集。

1.3.2 有监督学习

既然损失函数可以用来评价模型的好坏，那么让损失函数的值最小的那组参数就应该是最好的参数：

$$\hat{w} = \arg\min_{w} \text{loss}(w)$$

在数学上，这是一个无约束优化问题，即调整一组变量使得某个表达式最小（或最大），而对所调整的变量的取值没有限制。

但这里还有个问题，我们是否可以用测试集上的损失函数来调整参数呢？答案是否定的，这会产生严重的过拟合，即由于参数的学习是依赖于某个给定的样本及标准输出的集合，那么学习得到的模型就很有可能在这一集合上的损失很低，但在之外的集合上的损失就会偏高。那么如果用训练所用集合来评价模型，分数就会偏高。还有一个形象的比方是，考试是用来检验学生对知识的掌握，学生需要在考试之外也能运用知识。如果学生在学习时过分针对考试（甚至是知道考试的题目），那么可以想象这种应试的学习并不能保证学生在考试之外能够真正运用知识。

以上讨论了一大堆，解决方法其实并不复杂，即使用两个集合，一个是测试集，一个是训练集，参数的学习只针对训练集，找到使训练集损失尽量小的参数，然后在测试集上测试该组参数面对训练集之外的样本的表现。

1.3.3 梯度下降法

解决以上的无约束优化问题，就成了纯粹的应用数学问题。梯度下降法是解决这一类问题的基本方法，也被普遍用于神经网络参数的优化。

梯度下降法是一种迭代的方法。首先任意选取一组参数，然后一次次地对这组参数进行微小的调整，不断使得新的参数的损失函数更小。

这个方法可以这样形象地理解，不妨设需要优化的参数只有两个，则参数空间是一个二维的平面。任意一组参数对应于一个损失函数值，这构成第三维。这个三维的空间形成一个曲面，如同高低不平的地形图，经纬度表示参数，高度表示损失函数的值。那么优化问题就是找到高度最低的经纬度。梯度下降法的思路是，首先任意选择一个地点，然后在当前点找到坡度最陡的方向（例如一个坡面南低北高，则南北方向坡度大，东西方向坡度小），沿着该方向向下坡方向迈出一小步，作为新的地点进行下次迭代。这样不断地进行迈步，就可以走到一个海拔较低的地方。

所谓的坡度在数学上就是微积分中的梯度，梯度下降算法的形式化描述是：

梯度下降算法
1 初始化参数 W_0、t=0 2 步数 $t \leftarrow t+1$ 3 计算梯度 $\nabla W = \frac{\partial loss}{\partial W}\Big\|_{W_{t-1}}$ 4 更新参数 $W_t \leftarrow W_{t-1} - \eta \nabla W$ 5 如果收敛，结束并输出 W_t，否则转到步骤 2

这里主要的计算是第 3 步，计算梯度。这要求损失函数对于参数可导。

梯度下降法是一个可以用来处理任何非约束优化问题的方法，但它却不能彻底解决该问题。它最大的不足是无法保证最终得到全局最优的结果，即最终的结果通常并不能保证使损失函数最小，而只能保证在最终结果附近，没有更好的结果。因为梯度下降算法在更新参数的过程中，只利用了参数附近的梯度，对于整个参数空间的趋势，它是没有考虑的。

此外，如果真正尝试在计算机上实现梯度下降算法，会发现这个已经有理论缺陷的算法在实际使用中，有更多的问题限制了它的效果。例如，步长 η 的确定，如果太小，算法需要很长时间收敛，如果太大，又无法稳定到参数空间中的某一点。不同的设计会影响算法速度以及最终结果的好坏。但是到目前为止并没有理论上的好办法解决，因此步长的确定就从一个科学问题变成了工程问题。一种方法是让步长随着时间 t 的推移而变小，即在初期大步走，到后期小步挪。

下面这个例子展示了梯度下降算法（见图 1.5）可能犯的错误。

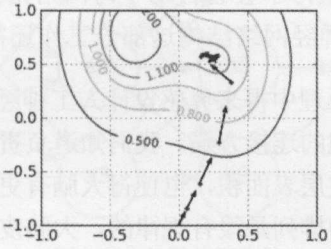


图 1.5 梯度下降算法

优化的目标函数是

$$1.2e^{-(x-0.8)^2-(y-0.5)^2}+e^{-(x+0.8)^2-(y-0.8)^2}$$

步长是 1.2。迭代从点（0, −1）开始，箭头指示的是梯度下降的方向。

在以上的图中可以看到：

（1）由于步长在开始时过小，图中开始时移动比较慢；

（2）更新比较盲目，没有全局的信息，往往是曲线的；

（3）后期步长过大，移动过快，在局部最优点周围震荡；

（4）最终不能收敛到全局最优点。

1.4 多层前馈网络

在前面我们已经介绍了人脑的神经细胞，以及由此抽象出来的人工神经网络模型，并且介绍了自动训练模型参数（即根据具体问题确定神经网络中边的权重）的方法。人工神

经元相互连接可以构成很多不同拓扑结构的神经网络。本节将介绍一种使用较为普遍的多层前馈网络，这也是深度学习中普遍使用的网络结构。

1.4.1 多层前馈网络

人工神经网络可以用图表示，神经元是图的节点，神经元之间的联系是图中的有向边。神经元不同的连接方式就会形成不同拓扑结构的神经网络。目前神经网络的自动学习几乎只局限于神经网络边连接的权重，不会去自动学习网络拓扑结构。而不同拓扑结构的网络适用于不同类型的问题，设计神经网络结构这部分工作还需要设计者的参与。

如同从人脑神经细胞工作原理中得来灵感设计人工神经元，在设计人工神经元连接方式时，也值得参考人脑中神经细胞的连接方式。我们知道负责高层信息处理的脑细胞集中在大脑皮层。人脑大量的沟回增大皮层表面积，也使得人脑有更强的信息处理能力。如果细看皮层，可以发现其中的神经细胞的排列是很有规律的，大脑皮层细胞的示意图如图 1.6 所示。

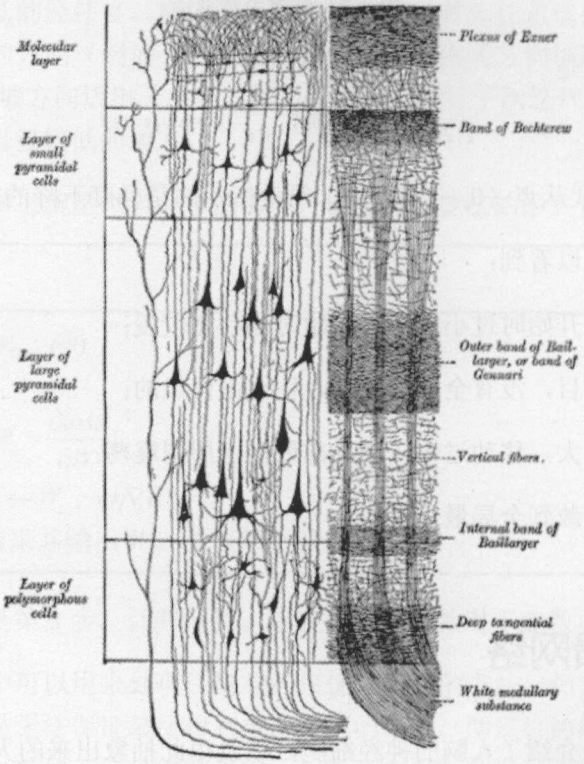


图 1.6 大脑皮层细胞示意图

图 1.6 是一幅大脑皮层的示意图，可以看到皮层中的神经细胞是分层排列的，如同汉堡或三明治。同时，树突和轴突也是有方向性的，即电信号都是由内层神经元传向外层神经元的。

与这种人脑皮层神经细胞连接方式类似的人工神经网络就是多层前馈网络。它的示意图如图 1.7 所示：

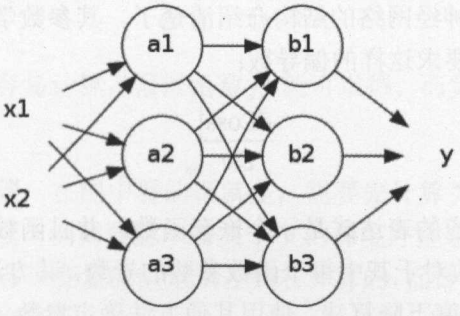


图 1.7 多层前馈网络

信号由左至右向前单向传播，每一列神经元构成一层，因此被称为多层前馈网络。网络相邻两层的任意神经元之间都有连接，此外没有层间连接或者跨越多层的连接。网络的输入/输出数目和形式根据具体问题指定，网络层数和每层神经元的个数也是需要人为指定的。

一个多层前馈网络将单个的、功能有限的人工神经元组织成了一个整体的计算单元，类似于计算机中的一个函数或一段程序。但它与我们熟悉的计算机程序又有所不同。它的计算没有迭代，并且是并行的。这很类似人脑进行视觉信息处理的情况。人脑的视觉皮层（visual cortex）负责处理视觉信息，视网膜得到一个影像后，大脑并不会一个“像素” 一个“像素”地进行扫描并处理，然后分辨看到的到底是什么，而是通过一组神经细胞，同时对传回的信号进行并行处理。这也是为什么擅长于串行计算的 CPU 不适合处理视频信息，电脑的图像处理需要长于并行计算的 GPU 来完成。有意思的是这也适用于人工神经网络，使用 GPU 来模拟人工神经网络要比使用 CPU 高效得多。

让我们回到图 1.6 所示的神经网。在数学上，它可以写成嵌套函数。图中有三列箭头，分别将前一层的数据变为后一层的数据，对应着三个函数，可用 f_1、f_2、f_3 表示：

$$
\begin{aligned}
a &= f_1(x) = s(xW_1 + b_1) \\
b &= f_2(a) = s(aW_2 + b_2) \\
y &= f_3(b) = s(bW_3 + b_3)
\end{aligned}
$$

那么整个函数就表示为：

$$y = f_1(f_2(f_3(x)))$$

1.4.2 后向传播算法计算梯度

至此我们已经将多层神经网络的结构介绍清楚了。其参数学习方法可以使用上节介绍的梯度下降算法，也就是要求这样的偏导数：

$$\left.\frac{\partial \text{Loss}}{\partial w}\right|_w$$

由于多层前馈网络对应的表达式是一个嵌套函数，并且函数的输入与输出都可能是向量，因此想要求出损失函数对于其中每个函数参数的导数，其方法并非那么直观。最初（几十年前），学者甚至绕过梯度下降算法，使用其他方法确定参数。直到计算此梯度的一般方法被提出，还被冠以一个单独的名字——“后向传播算法”。该算法使得可以使用梯度下降算法确定参数，因此更有效，被以里程碑的形式写入历史。这里我们不想细致介绍计算多层前馈神经网络中的每一个步骤，罗列每一个公式，这些内容读者在任何一本教材甚至许多网络文章中都可以找到（见延伸阅读部分）。在此我们只想说明后向传播算法所使用的数学原理，其实也就是每本微积分教材中会提到的计算导数的链式法则。

让我们回忆一下如何计算 $y=(2x+1)^2$ 的导数。我们学到的方法是先将 $2x+1$ 看作一个整体 $z=2x+1$，然后分别计算 z^2 和 $2x+1$ 的导数，然后相乘就可以了，即

$$\frac{dy}{dx} = \frac{dy}{dz}\frac{dz}{dx}$$

上面公式的物理意义就更加明显了。我们想要计算 $\frac{dy}{dx}$ 即 x 变化一个微小量后，y 的变化有多大。我们知道 x 会影响 z，然后 z 再影响 y。那么我们就可以先计算 x 的变化对 z 的影响有多大，然后计算 z 的变化对 y 的影响有多大，两者合起来（相乘）就可以得到 x 变化一个微小量后，y 的变化有多大。甚至更为直观的解释是直接将上式右边理解为两个分式相乘，消去相同项后就得到右式（如果将公式中的 “d” 换成 “Δ”，这样说就更为严谨了）。最早使用链式法则可能可以追溯到莱布尼茨。

回到前述的神经网络中，如果我们要计算 $\frac{\partial \text{loss}}{\partial w_2}$，实际上只需要将其分解为：

$$\frac{\partial \text{loss}}{\partial b}\frac{\partial b}{\partial w_2}$$

第二项比较容易，根据函数 f_2 的表达式就可以求得。而第一项是另一个关于变量 b 梯度的梯度，也可以被分解：

$$\frac{\partial \text{loss}}{\partial b}=\frac{\partial \text{loss}}{\partial \text{y}}\frac{\partial \text{y}}{\partial b}$$

以上第二项同样比较容易计算，根据函数 $f3$ 就可求得。而第一项可以根据具体损失函数求得其梯度。

这就是梯度计算的步骤，在图中看的话就是，需要先计算关于靠右变量的梯度，才能计算出关于靠左变量的梯度，例如要想求得关于 $\boldsymbol{y}$ 的梯度才能求关于 $\boldsymbol{b}$ 的梯度，最后才能求得关于 W_2 的梯度。计算各个变量的值是从左到右计算的，而计算梯度是从右向左计算的，这就是后向传播算法名字的来历。

以上我们看到，计算梯度所使用的后向传播算法，也就是微积分中最为普通的链式法则的具体应用。可见数学对于计算机科学也是非常重要的。

1.5 逐层预训练

到目前为止，之前所介绍的全部技术内容都是深度学习的主要组成部分，但是也都是在深度学习被提出之前早已有之的内容。本小节要介绍撑出深度学习这个大胖子的最后一个馒头——深度神经网络的预训练。

本节之前所介绍的深度神经网络，已经可以进行训练、解码等全部操作。并且很多年前，学者和工程师也都会如此使用。但这样训练得到的神经网络模型的效果却并不尽如人意。原因在上小节中已经涉及，即类似于随机梯度下降算法这样的参数优化算法，只能得到局部最优的解。而深度神经网络的参数空间更大，且前几层的参数的梯度会比较小，这都不利于算法收敛到全局较好的解。这直接导致了在同样的任务中使用相似的特征，神经网络的方法不如支持向量机（特别是由非线性核函数支持向量机）的效果好。

直到，Hinton 于 2006 年在美国 Science 杂志上发表了一篇与深度神经网络参数预训练有关的文章（Hinton & Salakhutdinov 2006），深度神经网络的预训练研究才有了些眉目。这种方法可以被称为逐层预训练。

在介绍这种预训练之前，让我们先来了解一种特殊的神经网络——自动编码器（auto-encoder）（Vincent, et al. 2008），它是一种只有一层隐层的神经网络，而且训练参数的目标很特殊——让输出尽量等于输入。但这并不是做了一件没有意义的事，因为当我们训练好整个网络的时候，我们在隐层得到了包含输入样本几乎全部无损信息的另一种表示形式，因为通过隐层到输出层的变换，我们能将其完全还原成输入。输入层到隐层的过程称为编码过程，隐层到输出层的过程称为解码过程。

可以想象，如果隐层神经元个数大于等于输入层，则只要简单地将输入层拷贝到隐层就可以达到这样的目的。因此，通常隐层神经元个数小于输入层，或者在隐层加入额外的限制。

在介绍了自动编码器之后，就可以给出一种使用自动编码器逐层训练的深度神经网络预训练方法。

首先，构造一个自动编码器神经网络，输入为深度神经网络的输入，隐层神经元个数为深度神经网络第一个隐层神经元个数。如此训练自动编码器后，记录自动编码器输入层到隐层的连接权重，这些权重会作为深度神经网络输入层到第一个隐层权重的初始值。

这样深度神经网络输入层到隐层的权重被预训练。此时构造另一个自动编码器神经网络，输入层是刚才深度神经网络的第一个隐层（其每个输入样本根据输入样本和刚才预训练的权重计算得到），隐层是深度神经网络的第二个隐层。仍然按照刚才的方式训练自动编码器神经网络，让第二隐层的信号能够尽量还原成第一隐层的信号。如此训练自动编码器后，记录自动编码器输入层到隐层的权重，作为深度神经网络第一个隐层到第二个隐层的权重。

如此逐层重复上面的步骤，直到所有层之间的权重均被初始化。最后用这些初始化后的权重，按照前一节的方式训练整个深度神经网络，这最后一步称为微调（fine tuning）。

经过预训练后得到的神经网络，会明显优于随机给出初始权重训练得到的神经网络。这个现象在其他的研究中也能得到了印证。

如果读者熟悉隐马尔可夫模型，就知道，经典的隐马模型第三个问题就是模型参数的训练问题。使用 EM 迭代的方法，得到的也是一个局部最优解，而且结果严重依赖于初始值。有一篇论文（Goldberg, et al. 2008）的题目是“EM Can Find Pretty Good HMM POS-Taggers（When Given a Good Start）”，就指出了这种模型训练初始值设定的重要性。

一个更为著名的模型是统计机器翻译中的 IBM 模型。要用随机的初始值直接训练一个效果好的复杂模型的参数几乎是不可能的。这个 IBM 模型是由一系列由简单到复杂的模型构成的，简单模型均是复杂模型的简化版本，参数也为复杂模型的一个子集。当要训练复杂模型

时，先训练简单模型，并用简单模型得到的参数作为训练复杂模型时那部分参数的初始值。

1.6 深度学习是终极神器吗

本章简单介绍了深度神经网络这一模型，它能完成什么功能，以及参数如何被学习。作为总结，本小节试图讨论它跟之前的模型比多做到了什么和仍然尚未做到什么。以此回答深度学习在多大深度上实现了“机器设计机器”这一终极神器的目标。

1.6.1 深度学习带来了什么

（1）强调了数据的抽象

对于我们心目中的智能机器，有抽象的能力是必须的。这也就是“深”与之前的“浅”的差别。深度学习通过多层的神经网络，将事物的表面特征变为更深的、更抽象的特征。

这与浅层学习的代表——支持向量机——相比，可以算是一个突破。最基本支持向量机是基于线性可分的。增加一层核函数后，其目的仍然只是使得样本在新空间线性可分。而核函数大多也只是手工设计，不能抽取深层的抽象概念。

（2）强调了特征的自动学习

智能的一个基本的特点是自动化。传统的机器学习算法已经让模型的设计更为自动化，通过数据可以自动确定模型的参数，只是模型的特征需要融入设计者的智慧手动设计。而深度学习的思想之一是特征也应该被自动地设计。人的智慧，可以只用来构造自动设计特征的方法。深度学习本身，就是这样的一种可以自动设计特征的方法。

深度学习让我们知道了特征自动设计的好处。这扩大了机器学习模型设计的可能性。现在学者们可以使用深度学习自动设计特征，当然也可以提出其他方法自动设计特征。

还有一点可以强调的是，这种特征的设计是可以不需要人工标注数据的，例如前面介绍的自动编码器，只需要输入样本的原始特征，不需要样本的标准输出（例如分类结果）或其他信息。这与半监督学习相仿，能够利用大量的廉价的数据帮助提高模型的效果。

（3）对连接主义的重视

深度神经网络对数据的抽象并不是从数据中提取出某种离散的级别、标签等。在这种

连接主义的观点下，所谓的抽象仍然是一些神经网络能够处理的实数向量。不过这也与大脑的运行机制相契合。

这种连接主义模型对于处理语音、图像信号感觉是理所当然的，语音、图像信号本来就是振幅、波长等物理量构成的向量、矩阵。

而连接主义某种看似激进的表现是将本来是符号表示的对象还原成向量形式的“信号”再进行处理。例如我们日常使用的文本，本来是被编码成相互独立的、每个都有各自意义的符号，而深度学习要将这些符号还原为向量再进行处理。实践表明：这种方法也是能够提高效果的，虽然没有在语音、图像上那么明显。

这种看似激进的做法仍然有合理性。因为我们的大脑在理解听到、看到的语言文字的时候，很可能也不是按照符号的方式处理的。

1.6.2 深度学习尚未做到什么

（1）缺少完善的理论

深度神经网络需要人设计的东西仍然很多：层数、每层的神经元个数，每次训练的步长。它们对模型影响怎样，并没有很好的理论基础作为参考。因此以下事情时有发生。

你看到很多论文，讲了很多设计模型的小技巧，模型的小变种。但其中的一部分不能被很好地得到理论上的解释。

你不断的调整需要手工设定的参数，得到一个很好的模型，但你不能解释为什么。

更为诡异的可能是，后来你发现你的代码中有个 bug，当改正后，得到了你真正想要的模型，但这个模型的效果却明显不如有 bug 的模型。

（2）缺少更为宏观的框架

深度网络与人脑相比，规模十分有限，最多也只能对应到皮层很小的一个区域。只能完成单一的任务，对应到人脑也就是一瞬间的事情。而如何去建模人脑更长期的机制（如记忆机制），以及如何使各个深度神经网络相互协同（如注意机制），仍然有待探索。

此外，人工智能中的某些重要问题，如常识如何引入，深度神经网络也暂时无法回答。

可见，深度学习在机器学习相关研究的某些方面提出了很有建设性的建议，但它还缺少理论基础，并且那些与人工智能相关的更多更关键的问题，还不是它能够解决的。

1.7 内容回顾与推荐阅读

本章主要以感知器神经网络为对象介绍了深度神经网络，此外基于受限玻尔兹曼机的深度网络也是主流方案（Bengio, et al. 2007）。本章以梯度下降法来介绍参数学习。其他流行的参数学习方法还有随机梯度下降法（SGD）、拟牛顿法（如 L-BFGS）等。本章以多层全连接网络为例介绍深度学习。常见的层间连接方式除了全连接外还有卷积（Convolutional Neural Network, CNN）。而层的组织形式除了多层叠之外常见的还有递归形式（Recursive Neural Network, RNN）和循环形式（Recurrent Neural Network, RNN）。

前面已经讨论过深度学习其实是一个没有太严格理论基础的体系，因此常常能看见一些不能完全说清楚道理但又十分有用的“奇技淫巧”，例如在神经网络的训练过程中人为随机破坏其中的信号（Srivastava, et al. 2014）。

另外深度学习在当前是一个发展十分迅速的方向，就在最近，一些看似深度学习的“标配”方案也在被颠覆。例如激活函数换作非 Sigmoid 函数且不再需要预训练（Zeiler, et al. 2013）。

以下再列出与 deep learning 相关的网络资源。

- http://www.cs.toronto.edu/～hinton/ 鼻祖 Hinton 的主页。
- http://ufldl.stanford.edu/wiki/index.php/UFLDL_Tutorial Andrew Ng 写的关于 deep learning 的 tutorial。其中还列举了一些推荐的学术参考文献（http://ufldl.stanford.edu/wiki/index.php/UFLDL_Recommended_Readings）。
- http://deeplearning.net/tutorial/ 另一个 Bengio 组的 tutorial，其中包含 Theano 的介绍。Theano 是 Python 写的可用于实现 deep learning 算法的工具包，提供自动推导梯度等功能，核心算法自动用 C 语言编译执行，且支持 GPU。此外 C++的 caffe（http://caffe.berkeleyvision.org/）、lua 的 torch(http://torch.ch/)都是常用的工具包。
- http://deeplearning.net/deep-learning-research-groups-and-labs/列举了与深度学习相关的研究机构。

1.8 参考文献

[1] （Raina, et al. 2009）Raina, R., Madhavan, A., & Ng, A. Y.（2009）. Large-scale deep unsupervised learning using graphics processors. In ICML（Vol. 9, pp. 873–880）.

[2] （Mnih & Hinton 2009）Mnih, A., & Hinton, G. E.（2009）. A scalable hierarchical distributed language model. In Advances in neural information processing systems（pp. 1081–1088）.

[3] （Socher, et al. 2013）Socher, R., Bauer, J., Manning, C. D., & Andrew Y., N.（2013）. Parsing with Compositional Vector Grammars. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics（Volume 1: Long Papers）（pp. 455–465）. Sofia, Bulgaria: Association for Computational Linguistics.

[4] （Socher, et al. 2012）Socher, R., Huval, B., Manning, C. D., & Ng, A. Y.（2012）. Semantic compositionality through recursive matrix-vector spaces. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning（pp. 1201–1211）.

[5] （Devlin, et al. 2014）Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R., & Makhoul, J.（2014）. Fast and robust neural network joint models for statistical machine translation. In 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, June.

[6] （Hinton & Salakhutdinov 2006）Hinton, G. E., & Salakhutdinov, R. R.（2006）. Reducing the dimensionality of data with neural networks. Science, 313（5786）, 504–507.

[7] （Vincent, et al. 2008）Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P.-A.（2008）. Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning（pp. 1096–1103）.

[8] （Goldberg, et al. 2008）Goldberg, Y., Adler, M., & Elhadad, M.（2008）. EM Can Find Pretty Good HMM POS-Taggers（When Given a Good Start）. In ACL（pp. 746–754）. Citeseer.

[9] （Bengio, et al. 2007）Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H.（2007）. Greedy layer-wise training of deep networks. Advances in neural information processing systems, 19, 153.

[10] （Zeiler, et al. 2013）Zeiler, M. D., Ranzato, M., Monga, R., Mao, M., Yang, K., Le, Q. V.（2013）. On rectified linear units for speech processing. In IEEE International Conference on Acoustics, Speech and Signal Processing（ICASSP）（pp. 3517–3521）. IEEE.

[11] （Srivastava, et al. 2014）Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R.（2014）. Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15（1）, 1929–1958.

第 2 章

知识图谱——机器大脑中的知识库

知识就是力量。

——[英]弗兰西斯·培根

2.1 什么是知识图谱

在互联网时代，搜索引擎是人们在线获取信息和知识的重要工具。当用户输入一个查询词，搜索引擎会返回它认为与这个关键词最相关的网页。从诞生之日起，搜索引擎就是这样的模式。

直到 2012 年 5 月，搜索引擎巨头谷歌在它的搜索页面中首次引入“知识图谱”：用户除了得到搜索网页链接外，还将看到与查询词有关的更加智能化的答案。如图 2.1 所示，当用户输入“Marie Curie”（玛丽·居里）这个查询词，谷歌会在右侧提供了居里夫人的详细信息，如个人简介、出生地点、生卒年月等，甚至还包括一些与居里夫人有关的历史人物，例如爱因斯坦、皮埃尔·居里（居里夫人的丈夫）等。

从杂乱的网页到结构化的实体知识，搜索引擎利用知识图谱能够为用户提供更具条理的信息，甚至顺着知识图谱可以探索更深入、广泛和完整的知识体系，让用户发现他们意想不到的知识。谷歌高级副总裁艾米特·辛格博士一语道破知识图谱的重要意义所在：“构成这个世界的是实体，而非字符串（things, not strings）”。

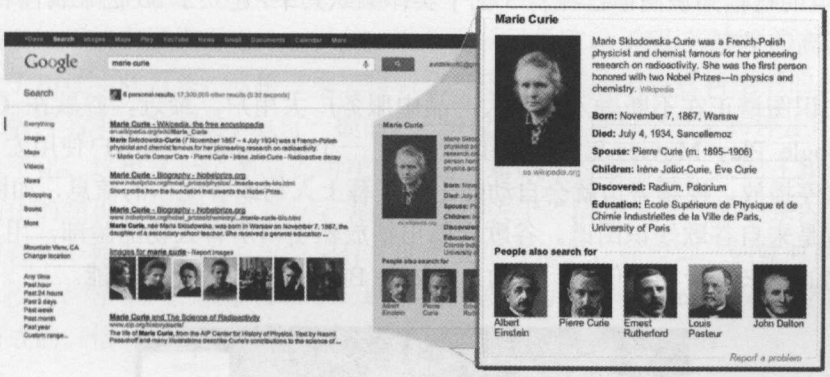


图 2.1 谷歌搜索引擎的知识图谱

谷歌知识图谱一出激起千层浪，美国的微软必应，中国的百度、搜狗等搜索引擎公司在短短的一年内纷纷宣布了各自的“知识图谱”产品，如百度“知心”、搜狗“知立方”等。为什么这些搜索引擎巨头纷纷跟进知识图谱，在这上面一掷千金，甚至把它视为搜索引擎的未来呢？这就需要从传统搜索引擎的原理讲起。以百度为例，在过去当我们想知道“泰山”的相关信息的时候，我们会在百度上搜索“泰山”，它会尝试将这个字符串与百度抓取的大规模网页做比对，根据网页与这个查询词的相关程度，以及网页本身的重要性，对网

页进行排序，作为搜索结果返回给用户。而用户所需的与“泰山”相关的信息，就还要他们自己动手，去访问这些网页来找了。

当然，与搜索引擎出现之前相比，随着网络信息的爆炸式增长，搜索引擎由于大大缩小了用户查找信息的范围，日益成为人们遨游信息海洋的不可或缺的工具。但是，传统搜索引擎的工作方式表明，它只是机械地比对查询词和网页之间的匹配关系，并没有真正理解用户要查询的到底是什么，远远不够“聪明”，当然经常会被用户嫌弃了。

而知识图谱则会将“泰山”理解为一个“实体”（entity），也就是一个现实世界中的事物。这样，搜索引擎会在搜索结果的右侧显示它的基本资料，例如地理位置、海拔高度、别名，以及百科链接等，此外甚至还会告诉你一些相关的“实体”，如嵩山、华山、衡山和恒山等其他三山五岳等。当然，用户输入的查询词并不见得只对应一个实体，例如当在谷歌中查询“apple”（苹果）时，谷歌不止展示 IT 巨头“Apple-Corporation”（苹果公司）的相关信息，还会在其下方列出“apple-plant”（苹果-植物）的另外一种实体的信息。

很明显，以谷歌为代表的搜索引擎公司希望利用知识图谱为查询词赋予丰富的语义信息，建立与现实世界实体的关系，从而帮助用户更快找到所需的信息。谷歌知识图谱不仅从 Freebase 和维基百科等知识库中获取专业信息，同时还通过分析大规模网页内容抽取知识。现在谷歌的这幅知识图谱已经将 5 亿个实体编织其中，建立了 35 亿个属性和相互关系，并还在不断高速扩充。

谷歌知识图谱正在不断融入其各大产品中服务广大用户。最近，谷歌在 Google Play Store 的 Google Play Movies & TV 应用中添加了一个新的功能，当用户使用安卓系统观看视频时，暂停播放，视频旁边就会自动弹出该屏幕上人物或者配乐的信息，如图 2.2 所示。这些信息就是来自谷歌知识图谱。谷歌会圈出播放器窗口所有人物的脸部，用户可以点击每一个人物的脸来查看相关信息。此前，Google Books 已经应用此功能。

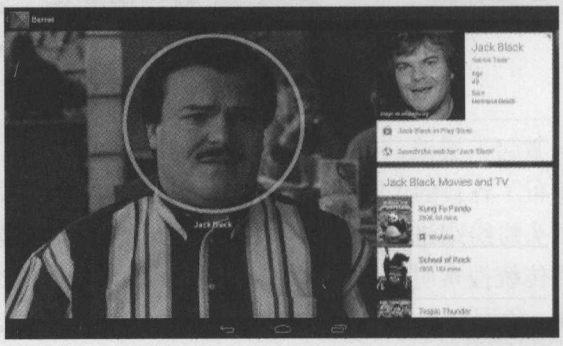


图 2.2 Google 利用知识图谱标示视频中的人物或配乐信息

2.2 知识图谱的构建

最初，知识图谱是由谷歌推出的产品名称，寓意与 Facebook 提出的社交图谱（Social Graph）异曲同工。由于其表意形象，现在知识图谱已经被用来泛指各种大规模知识库了。

我们应当如何构建知识图谱呢？我们先了解一下，知识图谱的数据来源都有哪些。知识图谱的最重要的数据来源之一是以维基百科、百度百科为代表的大规模知识库，在这些由网民协同编辑构建的知识库中，包含了大量结构化的知识，可以高效地转化到知识图谱中。此外，互联网的海量网页中也蕴藏了海量知识，虽然相对知识库而言这些知识更显杂乱，但通过自动化技术，也可以将其抽取出来构建知识图谱。接下来，我们分别详细介绍这些识图谱的数据来源。

2.2.1 大规模知识库

大规模知识库以词条作为基本组织单位，每个词条对应现实世界的某个概念，由世界各地的编辑者义务协同编纂内容。随着互联网的普及和 Web 2.0 理念深入人心，这类协同构建的知识库，无论是数量、质量还是更新速度，都早已超越传统由专家编辑的百科全书，成为人们获取知识的主要来源之一。目前，维基百科已经收录了超过 2200 万词条，而仅英文版就收录了超过 400 万条，远超过英文百科全书中最权威的大英百科全书的 50 万条，是全球浏览人数排名第 6 的网站。值得一提的是，2012 年大英百科全书宣布停止印刷版发行，全面转向电子化。这也从一个侧面说明在线大规模知识库的影响力。人们在知识库中贡献了大量结构化的知识。如图 2.3 所示，是维基百科关于“清华大学”的词条内容。可以看到，在右侧有一个列表，标注了与清华有关的各类重要信息，如校训、创建时间、校庆日、学校类型、校长，等等。在维基百科中，这个列表被称为信息框（infobox），是由编辑者们共同编辑而成的。信息框中的结构化信息是知识图谱的直接数据来源。

除了维基百科等大规模在线百科外，各大搜索引擎公司和机构还维护和发布了其他各类大规模知识库，例如谷歌收购的 Freebase，包含 3900 万个实体和 18 亿条实体关系；DBpedia 是德国莱比锡大学等机构发起的项目，从维基百科中抽取实体关系，包括 1 千万个实体和 14 亿条实体关系；YAGO 则是德国马克斯·普朗克研究所发起的项目，也是从维基百科和 WordNet 等知识库中抽取实体，到 2010 年该项目已包含 1 千万个实体和 1.2 亿条实体关系。此外，在众多专门领域还有领域专家整理的领域知识库。

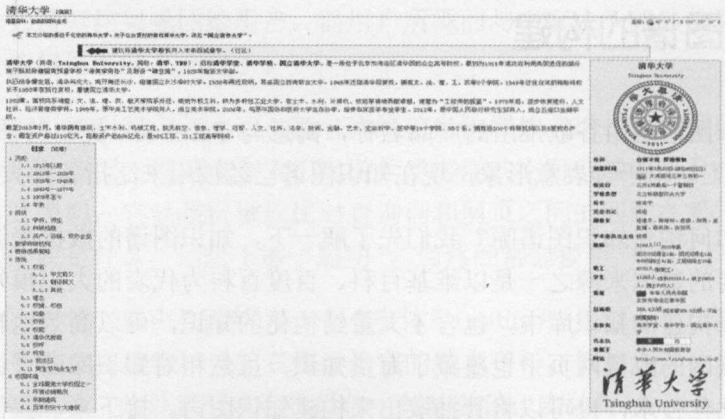


图 2.3 维基百科词条“清华大学”部分内容

2.2.2 互联网链接数据

国际万维网组织 W3C 在 2007 年发起了开放互联数据项目（Linked Open Data，LOD），其发布数据集示意图如图 2.4 所示。该项目旨在将由互联文档组成的万维网（Web of documents）扩展成由互联数据组成的知识空间（Web of data）。LOD 以 RDF（Resource Description Framework）形式在 Web 上发布各种开放数据集，RDF 是一种描述结构化知识的框架，它将实体间的关系表示为（实体 1，关系，实体 2）的三元组。LOD 还允许在不同来源的数据项之间设置 RDF 链接，实现语义 Web 知识库。目前世界各机构已经基于 LOD 标准发布了数千个数据集，包含数千亿 RDF 三元组。随着 LOD 项目的推广和发展，互联网会有越来越多的信息以链接数据形式发布，然而各机构发布的链接数据之间存在严重的异构和冗余等问题，如何实现多数据源的知识融合，是 LOD 项目面临的重要问题。

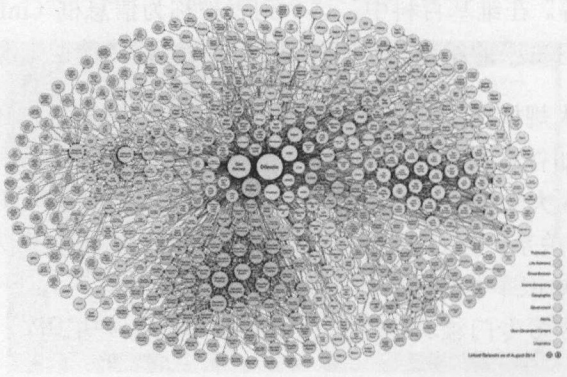


图 2.4 开放互联数据项目发布数据集示意图

2.2.3 互联网网页文本数据

与整个互联网相比，维基百科等知识库仍只能算沧海一粟。因此，人们还需要从海量互联网网页中直接抽取知识。与上述知识库的构建方式不同，很多研究者致力于直接从无结构的互联网网页中抽取结构化信息，如华盛顿大学 Oren Etzioni 教授主导的“开放信息抽取”（open information extraction，OpenIE）项目，以及卡耐基梅隆大学 Tom Mitchell 教授主导的“永不停止的语言学习”（never-ending language learning， NELL）项目。OpenIE 项目所开发的演示系统 TextRunner 已经从 1 亿个网页中抽取出了 5 亿条事实，而 NELL 项目也从 Web 中学习抽取了超过 5 千万条事实样例，如图 2.5 所示。

Recently-Learned Facts twitter Refresh

Instance	Iteration	date learned	confidence
moisturizing_lotion is a personal care product	770	16-sep-2013	92.7
santa_barbara_bromeliad_salamander is an amphibian	769	13-sep-2013	92.9
agra is a visualizable scene	771	26-sep-2013	100.0
long_eared_flying_squirrel is a mammal	766	04-sep-2013	94.0
samsung_2010 is a product	766	04-sep-2013	95.9
bras_lia is the capital city of the country brazil	771	26-sep-2013	96.9
saddam_hussein is a politician who holds the office of president	769	13-sep-2013	99.6
paul_stanley is a musician who plays the guitar	767	06-sep-2013	93.8
liz is a person who graduated from the university state_university	767	06-sep-2013	96.9
size is often found in second_bedroom	767	06-sep-2013	100.0

图 2.5 NELL 从 Web 中学习抽取事实样例

显而易见，与从维基百科中抽取的知识库相比，开放信息抽取从无结构网页中抽取的信息准确率还很低，其主要原因在于网页形式多样，噪声信息较多，信息可信度较低。因此，也有一些研究者尝试限制抽取的范围，例如只从网页表格等内容中抽取结构信息，并利用互联网的多个来源互相印证，从而大大提高抽取信息的可信度和准确率。当然这种做法也会大大降低抽取信息的覆盖面。天下没有免费的午餐，在大数据时代，我们需要在规模和质量之间寻找一个最佳的平衡点。

2.2.4 多数据源的知识融合

从以上数据来源进行知识图谱构建并非孤立地进行。在商用知识图谱构建过程中，需要实现多数据源的知识融合。以谷歌最新发布的 Knowledge Vault(Dong, et al. 2014)技术为例，其知识图谱的数据来源包括了文本、DOM Trees、HTML 表格、RDF 语义数据等多个来源。多来源数据的融合，能够更有效地判定抽取知识的可信性。

知识融合主要包括实体融合、关系融合和实例融合三类。对于实体，人名、地名、机

构名往往有多个名称。例如“中国移动通信集团公司”有“中国移动”、“中移动”、“移动通信”等名称。我们需要将这些不同名称规约到同一个实体下。同一个实体在不同语言、不同国家和地区往往会有不同命名，例如著名足球明星 Beckham 在大陆汉语中称作“贝克汉姆”，在香港译作“碧咸”，而在台湾则被称为“贝克汉”。与此对应的，同一个名字在不同语境下可能会对应不同实体，这是典型的一词多义问题，例如“苹果”有时是指一种水果，有时则指的是一家著名 IT 公司。在这样复杂的多对多对应关系中，如何实现实体融合是非常复杂而重要的课题。如前面开放信息抽取所述，同一种关系可能会有不同的命名，这种现象在不同数据源中抽取出的关系中尤其显著。与实体融合类似，关系融合对于知识融合至关重要。在实现了实体和关系融合之后，我们就可以实现三元组实例的融合。不同数据源会抽取出相同的三元组，并给出不同的评分。根据这些评分，以及不同数据源的可信度，我们就可以实现三元组实例的融合与抽取。

知识融合既有重要的研究挑战，又需要丰富的工程经验。知识融合是实现大规模知识图谱的必由之路。知识融合的好坏，往往决定了知识图谱项目的成功与否，值得任何有志于大规模知识图谱构建与应用的人士高度重视。

2.3 知识图谱的典型应用

知识图谱将搜索引擎从字符串匹配推进到实体层面，可以极大地改进搜索效率和效果，为下一代搜索引擎的形态提供了巨大的想象空间。知识图谱的应用前景远不止于此，目前知识图谱已经被广泛应用于以下几个任务中。

2.3.1 查询理解（Query Understanding）

谷歌等搜索引擎巨头之所以致力于构建大规模知识图谱，其重要目标之一就是能够更好地理解用户输入的查询词。用户查询词是典型的短文本（short text），一个查询词往往仅由几个关键词构成。传统的关键词匹配技术没有理解查询词背后的语义信息，查询效果可能会很差。

例如，对于查询词“李娜大满贯”，如果仅用关键词匹配的方式，搜索引擎根本不懂用户到底希望寻找哪个“李娜”，而只会机械地返回所有含有“李娜”这个关键词的网页。但通过利用知识图谱识别查询词中的实体及其属性，搜索引擎将能够更好地理解用户搜索意图。现在，我们到谷歌中查询“李娜大满贯”，会发现，首先谷歌会利用知识图谱在页面右

侧呈现中国网球运动员李娜的基本信息，我们可以知道这个李娜是指中国网球女运动员。同时，谷歌不仅像传统搜索引擎那样返回匹配的网页，更会直接在页面最顶端返回李娜赢得大满贯的次数“2”，如图 2.6 所示。

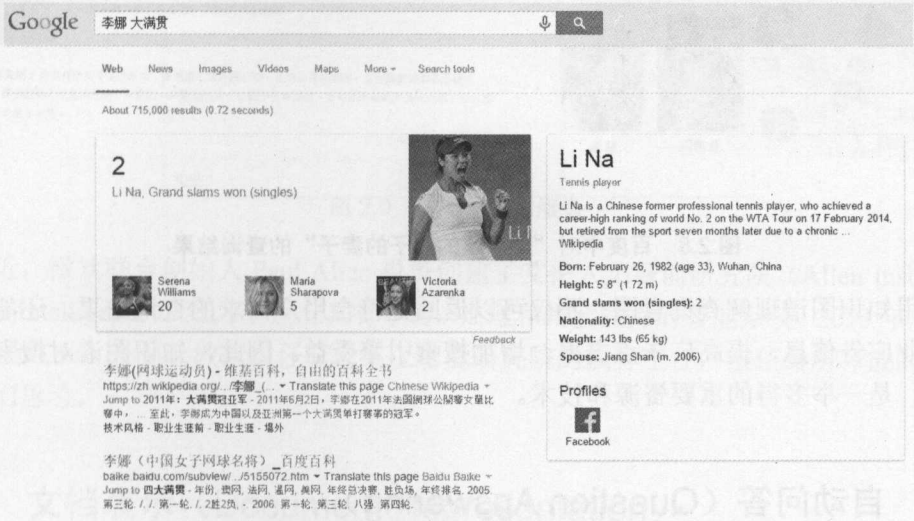


图 2.6 谷歌中对“李娜大满贯”的查询结果

主流商用搜索引擎基本都支持这种直接返回查询结果而非网页的功能，这背后都离不开大规模知识图谱的支持。以百度为例，图 2.7 是百度中对“珠穆朗玛峰高度”的查询结果，百度直接告诉用户珠穆朗玛峰的高度是 8844.43 米。

图 2.7 百度中对“珠穆朗玛峰高度”的查询结果

基于知识图谱，搜索引擎还能获得简单的推理能力。例如，图 2.8 是百度中对“梁启超的儿子的妻子”的查询结果，百度能够利用知识图谱知道梁启超的儿子是梁思成，梁思成的妻子是林徽因等人。

图 2.8 百度中对“梁启超的儿子的妻子”的查询结果

采用知识图谱理解查询意图，不仅可以返回更符合用户需求的查询结果，还能更好地匹配商业广告信息，提高广告点击率，增加搜索引擎受益。因此，知识图谱对搜索引擎公司而言，是一举多得的重要资源和技术。

2.3.2 自动问答（Question Answering）

人们一直在探索比关键词查询更高效的互联网搜索方式。很多学者预测，下一代搜索引擎将能够直接回答人们提出的问题，这种形式被称为自动问答。例如著名计算机学者、美国华盛顿大学计算机科学与工程系教授、图灵中心主任 Oren Etzioni 于 2011 年就在 Nature 杂志上发表文章“搜索需要一场变革“（Search Needs a Shake-Up）。该文指出，一个可以理解用户问题，从网络信息中抽取事实，并最终选出一个合适答案的搜索引擎，才能将我们带到信息获取的制高点。如上节所述，目前搜索引擎已经支持对很多查询直接返回精确答案而非海量网页而已。

关于自动问答，我们将有专门的章节介绍。这里，我们需要着重指出的是，知识图谱的重要应用之一就是作为自动问答的知识库。在搜狗推出中文知识图谱服务“知立方”的时候，曾经以回答“梁启超的儿子的太太的情人的父亲是谁？”这种近似脑筋急转弯似的问题作为案例，来展示其知识图谱的强大推理能力（搜狗知立方服务的实例如图 2.9 所示）。虽然大部分用户不会这样拐弯抹角地提问，但人们会经常需要寻找诸如“刘德华的妻子是谁？”、“侏罗纪公园的主演是谁？”、“姚明的身高？”以及“北京有几个区？”等问题的答案。而这些问题都需要利用知识图谱中实体的复杂关系推理得到。无论是理解用户查询意图，还是探索新的搜索形式，都毫无例外地需要进行语义理解和知识推理，而这都需要大规模、结构化的知识图谱的有力支持，因此知识图谱成为各大互联网公司的必争之地。

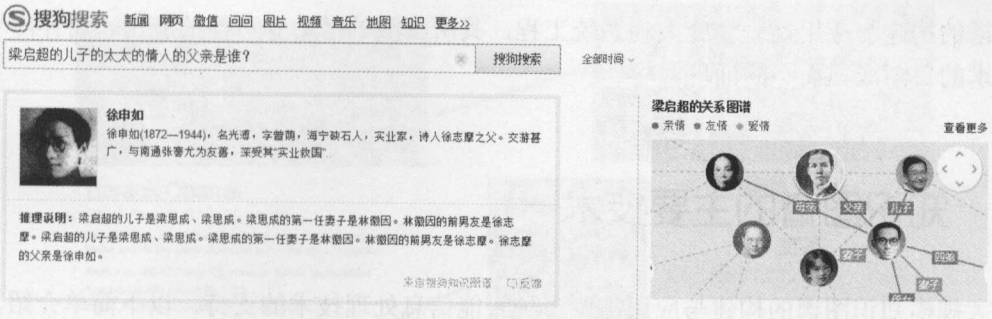


图 2.9 搜狗知立方服务

最近，微软联合创始人 Paul Allen 投资创建了艾伦人工智能研究院（Allen Institute for Artificial Intelligence），致力于建立具有学习、推理和阅读能力的智能系统。2013 年底，Paul Allen 任命 Oren Etzioni 教授担任艾伦人工智能研究院的执行主任，该任命所释放的信号颇值得我们思考。

2.3.3 文档表示（Document Representation）

经典的文档表示方案是空间向量模型（Vector Space Model），该模型将文档表示为词汇的向量，而且采用了词袋（Bag-of-Words，BOW）假设，不考虑文档中词汇的顺序信息。这种文档表示方案与上述的基于关键词匹配的搜索方案相匹配，由于其表示简单，效率较高，是目前主流搜索引擎所采用的技术。文档表示是自然语言处理很多任务的基础，如文档分类、文档摘要、关键词抽取，等等。

经典文档表示方案已经在实际应用中暴露出很多固有的严重缺陷，例如无法考虑词汇之间的复杂语义关系，无法处理对短文本（如查询词）的稀疏问题。人们一直在尝试解决这些问题，而知识图谱的出现和发展，为文档表示带来新的希望，那就是基于知识的文档表示方案。一篇文章不再只是由一组代表词汇的字符串来表示，而是由文章中的实体及其复杂语义关系来表示（Schuhmacher, et al. 2014）。该文档表示方案实现了对文档的深度语义表示，为文档深度理解打下基础。一种最简单的基于知识图谱的文档表示方案，可以将文档表示为知识图谱的一个子图（sub-graph），即用该文档中出现或涉及的实体及其关系所构成的图表示该文档。这种知识图谱的子图比词汇向量拥有更丰富的表示空间，也为文档分类、文档摘要和关键词抽取等应用提供了更丰富的可供计算和比较的信息。

知识图谱为计算机智能信息处理提供了巨大的知识储备和支持，将让现在的技术从基于字符串匹配的层次提升至知识理解层次。以上介绍的几个应用可以说只能窥豹一斑。知

识图谱的构建与应用是一个庞大的系统工程，其所蕴藏的潜力和可能的应用，将伴随着相关技术的日渐成熟而不断涌现。

2.4 知识图谱的主要技术

大规模知识图谱的构建与应用需要多种智能信息处理技术的支持，以下简单介绍其中若干主要技术。

2.4.1 实体链指（Entity Linking）

互联网网页，如新闻、博客等内容里涉及大量实体。大部分网页本身并没有关于这些实体的相关说明和背景介绍。为了帮助人们更好地了解网页内容，很多网站或作者会把网页中出现的实体链接到相应的知识库词条上，为读者提供更详尽的背景材料。这种做法实际上将互联网网页与实体之间建立了链接关系，因此被称为实体链指。

手工建立实体链接关系非常费力，因此如何让计算机自动实现实体链指，成为知识图谱得到大规模应用的重要技术前提。例如，谷歌等在搜索引擎结果页面呈现知识图谱时，需要该技术自动识别用户输入查询词中的实体并链接到知识图谱的相应节点上。

实体链指的主要任务有两个，实体识别（Entity Recognition）与实体消歧（Entity Disambiguation），都是自然语言处理领域的经典问题。

实体识别旨在从文本中发现命名实体，最典型的包括人名、地名、机构名等三类实体。近年来，人们开始尝试识别更丰富的实体类型，如电影名、产品名，等等。此外，由于知识图谱不仅涉及实体，还有大量概念（concept），因此也有研究者提出对这些概念进行识别。

不同环境下的同一个实体名称可能会对应不同实体，例如“苹果”可能指某种水果，某个著名IT公司，也可能是一部电影。这种一词多义或者歧义问题普遍存在于自然语言中。将文档中出现的名字链接到特定实体上，就是一个消歧的过程。消歧的基本思想是充分利用名字出现的上下文，分析不同实体可能出现在该处的概率。例如某个文档如果出现了iphone，那么“苹果”就有更高的概率指向知识图谱中的叫“苹果”的IT公司。

实体链指并不局限于文本与实体之间，如图2.10所示，还可以包括图像、社交媒体等数据与实体之间的关联。可以看到，实体链指是知识图谱构建与应用的基础核心技术。

图 2.10 实体链指实现实体与文本、图像、社交媒体等数据的关联

2.4.2 关系抽取（Relation Extraction）

构建知识图谱的重要来源之一是从互联网网页文本中抽取实体关系。关系抽取是一种典型的信息抽取任务。

典型的开放信息抽取方法采用自举（bootstrapping）的思想，按照“模板生成=>实例抽取”的流程不断迭代直至收敛。例如，最初可以通过“X 是 Y 的首都”模板抽取出（中国，首都，北京）、（美国，首都，华盛顿）等三元组实例；然后根据这些三元组中的实体对“中国-北京”和“美国-华盛顿”可以发现更多的匹配模板，如“Y 的首都是 X”、“X 是 Y 的政治中心”等等；进而用新发现的模板抽取更多新的三元组实例，通过反复迭代不断抽取新的实例与模板。这种方法直观有效，但也面临很多挑战性问题，如在扩展过程中很容易引入噪声实例与模板，出现语义漂移现象，降低抽取准确率。研究者针对这一问题提出了很多解决方案：提出同时扩展多个互斥类别的知识，例如同时扩展人物、地点和机构，要求一个实体只能属于一个类别；也有研究提出引入负实例来限制语义漂移。

我们还可以通过识别表达语义关系的短语来抽取实体间关系。例如，我们通过句法分析，可以从文本中发现“华为”与“深圳”的如下关系：（华为，总部位于，深圳）、（华为，总部设置于，深圳）、以及（华为，将其总部建于，深圳）。通过这种方法抽取出的实体间关系非常丰富而自由，一般是一个以动词为核心的短语。该方法的优点是，我们无需预先

人工定义关系的种类，但这种自由度带来的代价是，关系语义没有归一化，同一种关系可能会有多种不同的表示。例如，上述发现的“总部位于”、“总部设置于”以及“将其总部建于”等三个关系实际上是同一种关系。如何对这些自动发现的关系进行聚类归约是一个挑战性问题。

我们还可以将所有关系看做分类标签，把关系抽取转换为对实体对的关系分类问题。这种关系抽取方案的主要挑战在于缺乏标注语料。2009 年斯坦福大学的研究者提出远程监督（Distant Supervision）思想，使用知识图谱中已有的三元组实例启发式地标注训练语料。远程监督思想的假设是，每个同时包含两个实体的句子，都表述了这两个实体在知识库中的对应关系。例如，根据知识图谱中的三元组实例（苹果，创始人，乔布斯）和（苹果，CEO，库克），我们可以将以下四个包含对应实体对的句子分别标注为包含“创始人”和“CEO”关系：

样例	句子	关系/分类标签
苹果-乔布斯	苹果公司的创始人是乔布斯。	创始人
苹果-乔布斯	乔布斯创立了苹果公司。	创始人
苹果-库克	苹果公司的 CEO 是库克。	CEO
苹果-库克	库克现在是苹果公司的 CEO。	CEO

我们将知识图谱三元组中每个实体对看做待分类样例，将知识图谱中实体对关系看做分类标签。通过从出现该实体对的所有句子中抽取特征，我们可以利用机器学习分类模型（如最大熵分类器、SVM 等）构建信息抽取系统。对于任何新的实体对，根据所出现该实体对的句子中抽取的特征，我们就可以利用该信息抽取系统自动判断其关系。远程监督能够根据知识图谱自动构建大规模标注语料库，因此取得了瞩目的信息抽取效果。

与自举思想面临的挑战类似，远程监督方法会引入大量噪声训练样例，严重损害模型准确率。例如，对于（苹果，创始人，乔布斯）我们可以从文本中匹配以下四个句子：

句子	关系/分类标签	是否正确
苹果公司的创始人是乔布斯。	创始人	正确
乔布斯创立了苹果公司。	创始人	正确
乔布斯回到了苹果公司。	创始人	错误
乔布斯曾担任苹果的 CEO。	创始人	错误

在这四个句子中，前两个句子的确表明苹果与乔布斯之间的创始人关系；但是，后两个句子则并没有表达这样的关系。很明显，由于远程监督只能机械地匹配出现实体对的句子，因此会大量引入错误训练样例。为了解决这个问题，人们提出了很多去除噪声实例的办法，来提升远程监督性能。例如，研究发现，一个正确训练实例往往位于语义一致的区域，也就是其周边的实例应当拥有相同的关系；也有研究提出利用因子图、矩阵分解等方法，建立数据内部的关联关系，有效实现降低噪声的目标。

关系抽取是知识图谱构建的核心技术，它决定了知识图谱中知识的规模和质量。关系抽取是知识图谱研究的热点问题，还有很多挑战性问题需要解决，包括提升从高噪声的互联网数据中抽取关系的鲁棒性，扩大抽取关系的类型与抽取知识的覆盖面，等等。

2.4.3 知识推理（Knowledge Reasoning）

推理能力是人类智能的重要特征，能够从已有知识中发现隐含知识。推理往往需要相关规则的支持，例如从“配偶”+“男性”推理出“丈夫”，从“妻子的父亲”推理出“岳父”，从出生日期和当前时间推理出年龄，等等。

这些规则可以通过人们手动总结构建，但往往费时费力，人们也很难穷举复杂关系图谱中的所有推理规则。因此，很多人研究如何自动挖掘相关推理规则或模式。目前主要依赖关系之间的同现情况，利用关联挖掘技术来自动发现推理规则。

实体关系之间存在丰富的同现信息。如图 2.11 所示，在康熙、雍正和乾隆三个人物之间，我们有（康熙，父亲，雍正）、（雍正，父亲，乾隆）以及（康熙，祖父，乾隆）三个实例。根据大量类似的实体 X、Y、Z 间出现的（X，父亲，Y）、（Y，父亲，Z）以及（X，祖父，Z）实例，我们可以统计出“父亲+父亲=>祖父”的推理规则。类似地，我们还可以根据大量（X，首都，Y）和（X，位于，Y）实例统计出“首都=>位于”的推理规则，根据大量（X，总统，美国）和（X，是，美国人）统计出“美国总统=>是美国人”的推理规则。

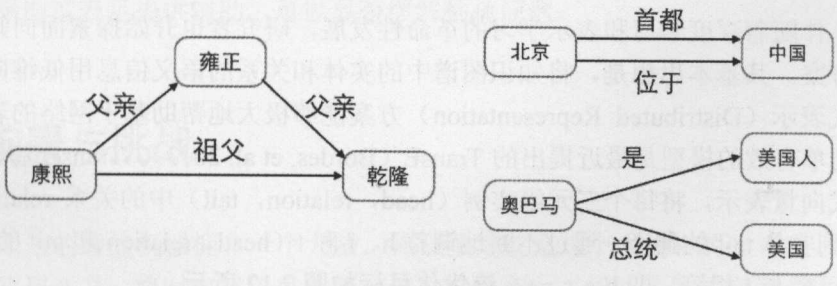


图 2.11 知识推理举例

知识推理可以用于发现实体间新的关系。例如，根据“父亲+父亲=>祖父”的推理规则，如果两实体间存在“父亲+父亲”的关系路径，我们就可以推理它们之间存在“祖父”的关系。利用推理规则实现关系抽取的经典方法是 Path Ranking Algorithm（Lao &Cohen 2010），该方法将每种不同的关系路径作为一维特征，通过在知识图谱中统计大量的关系路径构建关系分类的特征向量，建立关系分类器进行关系抽取，取得不错的抽取效果，成为近年来的关系抽取的代表方法之一。但这种基于关系的同现统计的方法，面临严重的数据稀疏问题。

在知识推理方面还有很多的探索工作，例如采用谓词逻辑（Predicate Logic）等形式化方法和马尔科夫逻辑网络（Markov Logic Network）等建模工具进行知识推理研究。目前来看，这方面研究仍处于百家争鸣阶段，大家在推理表示等诸多方面仍未达成共识，未来路径有待进一步探索。

2.4.4 知识表示（Knowledge Representation）

在计算机中如何对知识图谱进行表示与存储，是知识图谱构建与应用的重要课题。

如“知识图谱”字面所表示的含义，人们往往将知识图谱作为复杂网络进行存储，这个网络的每个节点带有实体标签，而每条边带有关系标签。基于这种网络的表示方案，知识图谱的相关应用任务往往需要借助于图算法来完成。例如，当我们尝试计算两实体之间的语义相关度时，我们可以通过它们在网络中的最短路径长度来衡量，两个实体距离越近，则越相关。而面向“梁启超的儿子的妻子”这样的推理查询问题时，则可以从“梁启超”节点出发，通过寻找特定的关系路径“梁启超->儿子->妻子->?”，来找到答案。

然而，这种基于网络的表示方法面临很多困难。首先，该表示方法面临严重的数据稀疏问题，对于那些对外连接较少的实体，一些图方法可能束手无策或效果不佳。此外，图算法往往计算复杂度较高，无法适应大规模知识图谱的应用需求。

最近，伴随着深度学习和表示学习的革命性发展，研究者也开始探索面向知识图谱的表示学习方案。其基本思想是，将知识图谱中的实体和关系的语义信息用低维向量表示，这种分布式表示（Distributed Representation）方案能够极大地帮助基于网络的表示方案。其中，最简单有效的模型是最近提出的 TransE（Bordes, et al. 2013）。TransE 基于实体和关系的分布式向量表示，将每个三元组实例（head，relation，tail）中的关系 relation 看做从实体 head 到实体 tail 的翻译，通过不断地调整 h、r 和 t（head、relation 和 tail 的向量），使（h + r）尽可能与 t 相等，即 h + r = t。该优化目标如图 2.12 所示。

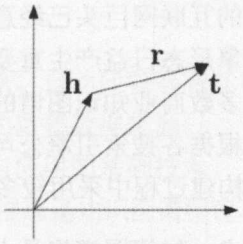


图 2.12 基于分布式表示的知识表示方案

通过 TransE 等模型学习得到的实体和关系向量，能够在很大程度上缓解基于网络表示方案的稀疏性问题，应用于很多重要任务中。

首先，利用分布式向量，我们可以通过欧氏距离或余弦距离等方式，很容易地计算实体间、关系间的语义相关度。这将极大地改进开放信息抽取中实体融合和关系融合的性能。通过寻找给定实体的相似实体，还可用于查询扩展和查询理解等应用。

其次，知识表示向量可以用于关系抽取。以 TransE 为例，由于我们的优化目标是让 h+r=t，因此，当给定两个实体 h 和 t 的时候，我们可以通过寻找与 t-h 最相似的 r，来寻找两实体间的关系。（Bordes, et al. 2013）中的实验证明，该方法的抽取性能较高。而且我们可以发现，该方法仅需要知识图谱作为训练数据，不需要外部的文本数据，因此这又称为知识图谱补全（Knowledge Graph Completion），与复杂网络中的链接预测（Link Prediction）类似，但是要复杂得多，因为在知识图谱中每个节点和连边上都有标签（标记实体名和关系名）。

最后，知识表示向量还可以用于发现关系间的推理规则。例如，对于大量 X、Y、Z 间出现的（X，父亲，Y）、（Y，父亲，Z）以及（X，祖父，Z）实例，我们在 TransE 中会学习 X+父亲=Y，Y+父亲=Z，以及 X+祖父=Z 等目标。根据前两个等式，我们很容易得到 X+父亲+父亲=Z，与第三个公式相比，就能够得到“父亲+父亲=>祖父”的推理规则。前面我们介绍过，基于关系的同现统计学习推理规则的思想，存在严重的数据稀疏问题。如果利用关系向量表示提供辅助，可以显著缓解稀疏问题。

2.5 前景与挑战

如果未来的智能机器拥有一个大脑，知识图谱就是这个大脑中的知识库，对于大数据智能具有重要意义，将对自然语言处理、信息检索和人工智能等领域产生深远影响。

现在以商业搜索引擎公司为首的互联网巨头已经意识到知识图谱的战略意义，纷纷投入重兵布局知识图谱，并对搜索引擎形态日益产生重要的影响。同时，我们也强烈地感受到，知识图谱还处于发展初期，大多数商业知识图谱的应用场景非常有限，例如搜狗知立方更多聚焦在娱乐和健康等领域。根据各搜索引擎公司提供的报告来看，为了保证知识图谱的准确率，仍然需要在知识图谱构建过程中采用较多的人工干预。

可以看到，在未来的一段时间内，知识图谱将是大数据智能的前沿研究问题，有很多重要的开放性问题亟待学术界和产业界协力解决。我们认为，未来知识图谱研究有以下几个重要挑战。

1. 知识类型与表示。知识图谱主要采用（实体 1，关系，实体 2）三元组的形式来表示知识，这种方法可以较好地表示很多事实性知识。然而，人类知识类型丰富多样，面对很多复杂知识，三元组就束手无策了。例如，人们的购物记录信息，新闻事件等，包含大量实体及其之间的复杂关系，更不用说人类大量的涉及主观感受、主观情感和模糊的知识了。有很多学者针对不同场景设计了不同的知识表示方法。知识表示是知识图谱构建与应用的基础，如何合理设计表示方案，更好地涵盖人类不同类型的知识，是知识图谱的重要研究问题。最近认知领域关于人类知识类型的探索（Tenenbaum, et al. 2011）也许会对知识表示研究有一定启发作用。

2. 知识获取。如何从互联网大数据萃取知识，是构建知识图谱的重要问题。目前已经提出各种知识获取方案，并已经成功抽取出大量有用的知识。但在抽取知识的准确率、覆盖率和效率等方面，都仍不尽如人意，有极大的提升空间。

3. 知识融合。从不同来源数据中抽取的知识可能存在大量噪声和冗余，或者使用了不同的语言。如何将这些知识有机融合起来，建立更大规模的知识图谱，是实现大数据智能的必由之路。

4. 知识应用。目前大规模知识图谱的应用场景和方式还比较有限，如何有效实现知识图谱的应用，利用知识图谱实现深度知识推理，提高大规模知识图谱计算效率，需要人们不断锐意发掘用户需求，探索更重要的应用场景，提出新的应用算法。这既需要丰富的知识图谱技术积累，也需要对人类需求的敏锐感知，找到合适的应用之道。

2.6 内容回顾与推荐阅读

本章系统地介绍了知识图谱的产生背景、数据来源、应用场景和主要技术。通过本章

我们主要有以下结论：

- 知识图谱是下一代搜索引擎、自动问答等智能应用的基础设施。
- 互联网大数据是知识图谱的重要数据来源。
- 知识表示是知识图谱构建与应用的基础技术。
- 实体链指、关系抽取和知识推理是知识图谱构建与应用的核心技术。知识图谱与本体（Ontology）和语义网（Semantic Web）等密切相关，有兴趣的读者可以搜索与之相关的文献阅读。知识表示（Knowledge Representation）是人工智能的重要课题，读者可以通过人工智能专著（Russell & Norvig 2009）了解其发展历程。在关系抽取方面，读者可以阅读（Nauseates, et al. 2013）、（Nickel, et al. 2015）详细了解相关技术。

2.7 参考文献

[1] (Bordes, et al. 2013) Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In Proceedings of NIPS.

[2] (Dong, et al. 2014) Dong, X., Gabrilovich, E., Heitz, G., Horn, W., et al. Knowledge Vault A web-scale approach to probabilistic knowledge fusion. In Proceedings of KDD.

[3] (Lao & Cohen 2010) Lao, N., & Cohen, W. W. (2010). Relational retrieval using a combination of path-constrained random walks. Machine learning, 81(1), 53-67.

[4] (Nauseates, et al. 2013) Nastase, V., Nakov, P., Seaghdha, D. O., & Szpakowicz, S. (2013). Semantic relations between nominals. Synthesis Lectures on Human Language Technologies, 6(1), 1-119.

[5] (Nickel, et al. 2015) Nickel, M., Murphy, K., Tresp, V., & Gabrilovich, E. A Review of Relational Machine Learning for Knowledge Graphs.

[6] (Russell & Norvig 2009) Russell, S., & Norvig, P. (2009). Artificial Intelligence: A Modern Approach, 3rd Edition. Pearson Press.（中文译名：人工智能——一种现代方法）.

[7] (Schuhmacher, et al. 2014) Schuhmacher, M., & Ponzetto, S. P. Knowledge-based graph document modeling. In Proceedings of the 7th ACM international conference on Web search and data mining. In Proceedings of WSDM.

[8] (Tenenbaum, et al. 2011) Tenenbaum, J. B., Kemp, C., Griffiths, T. L., & Goodman, N. D. (2011). How to grow a mind: Statistics, structure, and abstraction. science, 331(6022), 1279-1285.

第 3 章

大数据系统——大数据背后的支撑技术

工欲善其事，必先利其器。

——孔子

3.1 概述

从计算机系统的角度来看，大数据问题指的是那些数据规模大到一定的程度，使用传统技术无法或者很难处理的问题。大数据处理的应用和算法，最终都需要在计算机系统的支持下进行存储和计算，数据规模的爆炸性增长给数据处理系统的设计和实现提出了巨大的挑战。本章将向读者介绍大数据系统的前沿技术，使读者初步具备针对问题的特点和规模选取合适的大数据系统解决方案的能力。

数据的获取、存储、传输、分析、检索等过程一直是人类生产和生活中无时无刻都在进行的环节。随着科技的进步，尤其是互联网的飞速发展，我们所面临的问题规模不断增长。举例来说，人类的 DNA 有 30 亿左右个碱基对，每个碱基对可以用 2 个二进制位（bit，缩写：b）来表示，4 个碱基对就是 1 个字节（byte，缩写：B），每个人的基因就有 0.75GB 左右的数据量；哈勃望远镜每天都要向地球传回 10TB 左右的数据；截至 2014 年底，互联网上公开可见的网页就已经超过 45 亿个，Google 搜索引擎的索引数据超过 100PB。全世界人均存储容量也遵循类似于集成电路中的摩尔定律规则，自 20 世纪 80 年代以来，几乎每 40 个月就翻一翻。2012 年，全世界平均每天产生的数据量约为 2.5EB；而到了 2014 年，平均每天产生的数据量达到 2.3ZB。这里用的单位词头对照表如表 3.1 所示。

表 3.1 单位词头对照表

词头符号	十进制用法	二进制用法	数量
k	10^3	2^{10}	千
M	10^6	2^{20}	百万
G	10^9	2^{30}	十亿
T	10^{12}	2^{40}	万亿
P	10^{15}	2^{50}	千万亿
E	10^{18}	2^{60}	百亿亿
Z	10^{21}	2^{70}	十万亿亿
Y	10^{24}	2^{80}	亿亿亿

与此同时，中央处理器（Central Processing Unit，CPU）作为计算机内部进行控制和运算的核心部件，它的时钟频率也在不断提高，运算速度变快，成本降低。然而，经过二十

多年的快速发展，芯片的时钟频率遇到了物理极限的挑战。一是数据在芯片内的传输有一定的延迟，当延迟时间与一个时钟周期相当时，数据传输就会不稳定，从而导致芯片工作异常。二是高时钟频率的芯片在工作时会散发大量的热量，而芯片的表面积有限，如果热量不能及时散发，也会导致芯片无法正常工作。进入二十一世纪之后，像英特尔、AMD这些主流的芯片制造厂商逐渐放弃提高时钟频率，转而生产多核芯片，也就是利用芯片集成度还可以不断提高的工艺进步，将多个处理器单元集成到同一块芯片上。

传统算法研究的是可以在单处理器系统上运行的串行算法。然而要在这样的多核系统，甚至是由多台多核计算机通过网络连接构成的多机系统上并发地运行这些算法，除了算法核心模块在每个处理器上要能高效运行之外，如何对计算任务合理地进行划分，以及在计算过程中各个处理器之间通信的开销也会影响算法整体的效率。同时，多个处理器也意味着更容易发生硬件本身的故障，如何应对这些故障也是并行计算和分布式计算需要解决的问题。由此可见，相比于单处理器的程序而言，多核、多机的程序编写和调试的难度更大，更加需要底层操作系统和计算框架的支持。

下面，我们将分节向读者介绍大数据处理的相关技术，包括传统的高性能计算技术、虚拟化和云计算技术和以Google为代表的分布式计算技术及常见的开源实现版本Hadoop。另外，我们也将介绍更加注重计算性能的内存计算和大数据分析的重要应用场景图计算。同时，在存储方面，我们将介绍NoSQL的含义和分类。希望通过学习这些知识，读者能对大数据系统有一个总体的了解，并能初步根据自己所遇到的问题规模和特点来选取合适的解决方案。

3.2 高性能计算技术

高性能计算（High-Performance Computing，HPC）技术一直是计算机技术发展的前沿方向，从世界上第一台计算机ENIAC于1946年诞生开始，在一些重要应用（例如导弹弹道轨迹模拟、核爆模拟）的驱动下，高性能计算技术不断发展。TOP500网站每半年都会公布一次全球最快的计算机排名，在最新一期排名中，我国天河二号以33.863PFLOPS（每秒33.863×10^{15}次浮点运算操作）的性能排在世界第一位，表3.2列出了目前排名前5位的超级计算机。由于这些计算机系统的性能相比于我们日常使用的计算机而言要高出很多个数量级，因此也被称为超级计算机（supercomputer）。除了上述提到的军事用途的应用外，高性能计算机还用来计算天气预报、气候变化、油气勘探、分子模型等民用和科研方面的应用。

表 3.2 TOP500 前 5 位的超级计算机（2015 年 11 月）

排名	名称	峰值速度（PFLOPS）	制造者	国家
1	天河二号	33.863	国防科大	中国
2	Titan	17.590	Cray 公司	美国
3	Sequoia	17.173	IBM	美国
4	K computer	10.510	富士通	日本
5	Mira	8.586	IBM	美国

3.2.1 超级计算机的组成

早期的超级计算机只有单个处理器。到了 20 世纪 80 年代，出现了多个处理器的超级计算机。到了 90 年代，美国和日本开始制造有上千个处理器组成的超级计算机，通过复杂的高速网络将这些处理器连接起来。

现在的超级计算机由成千上万个节点组成，每个节点都是一台独立的计算机。因此，所谓的超级计算机其实并不是指一台具体的机器，而是指一套完整的系统，包括所有节点以及它们之间互联的网络设备，另外还有为了保障这些设备正常运行而配置的电源、冷却等装置。超级计算机的节点一般可分为计算节点、存储节点和管理节点。

- 计算节点主要负责计算，一般会配置多个多核处理器和较大的内存，还会配置 GPU、FPGA 等计算加速器。一般会带一些本地磁盘，用于存储计算中间结果。
- 存储节点负责数据的存储，包括原始数据和最终的计算结果，也有可能包含一部分的计算中间结果。
- 管理节点用于节点和用户的管理，负责对所有节点的运行状态进行监控，发生异常时会向管理员发出警告信息。同时，需要做计算的用户也会登录到管理节点上，通过任务调度系统提交计算任务，这一功能有时候也会由单独的登录节点来承担。

在当前的 TOP500 排名中，绝大多数超级计算机使用的都是 x86 体系结构的节点，有的还配置了 GPU 等加速器，用于提高性能功耗比。操作系统一般都使用 Linux。

以天河二号为例，它由 16000 个计算节点组成，每个节点有两颗 Xeon E5-2692 处理器和三块 Xeon Phi 加速卡，总计 312 万个核。每个计算节点配置 64GB 内存，每块 Xeon Phi 加速卡有 8GB 内存，总计 88GB。全部 16000 个计算节点共有 1.408PB 内存，而存储方面

则共有 12.4PB 的空间。节点之间采用自主研发的 Express-2 内部互联网络，传输速率为 6.36GB/s，延迟 85μs。这样的系统功耗也非常高，加上冷却系统之后天河二号的系统整体功耗为 24MW。

3.2.2 并行计算的系统支持

编写并行计算的程序需要进行创建线程、共享资源的竞争保护、多机之间的通信等操作。为了方便应用层的程序员编写此类程序，操作系统和第三方的软件库提供了这些功能。常见的并行编程组件有 Pthreads、OpenMP、MPI 等。

1. Pthreads

Pthreads（POSIX threads）是一个操作系统的标准，它定义了一套创建和操作线程的应用编程接口（Application Programming Interface，API），同时也包括互斥锁、条件变量等用于保护共享资源、实现线程间同步的工具，用于在同一台机器上编写运行多线程程序。每一个线程可以在单独的一个处理器核上运行，线程之间共享内存。

常见的 UNIX 系列的操作系统，例如 Linux、Mac OS X、FreeBSD 等，都提供了 Pthreads 的实现，而 Windows 下也有基于 Windows API 的第三方 Pthreads 库。

使用 Pthreads 编写多线程程序时，与正常的串行程序一样，刚开始运行时程序还是单线程的，这个线程一般被称为主线程。用户可以通过调用 pthread_create 函数来创建新的线程，调用时需要给出新线程运行哪个函数，以及相关的参数。pthread_create 函数调用成功后，程序就产生了一个新的线程（子线程），开始执行给出的函数。同时，原来的主线程还会继续执行。主线程可以通过 pthread_join 来等待子线程结束，并回收子线程占用的系统资源。

多个线程同时操作同一项系统资源可能会产生竞争。例如，有多个线程要对同一个计数器 C 进行增一操作，就有可能会产生如图 3.1 所示的竞争，发生逻辑上的错误。计数器增一操作要分 3 步执行，第一步是把计数器的值放入 CPU 的寄存器（在图中用 Reg 表示）中，第二步对寄存器中的值加一，第三步再把寄存器的值写回计数器。由于两个线程之间指令可能是任何顺序执行的，图中左边的情况就是一种可能的执行顺序，两个线程分别对 C 进行增一，故得到的最终结果是 1，而不是 2。

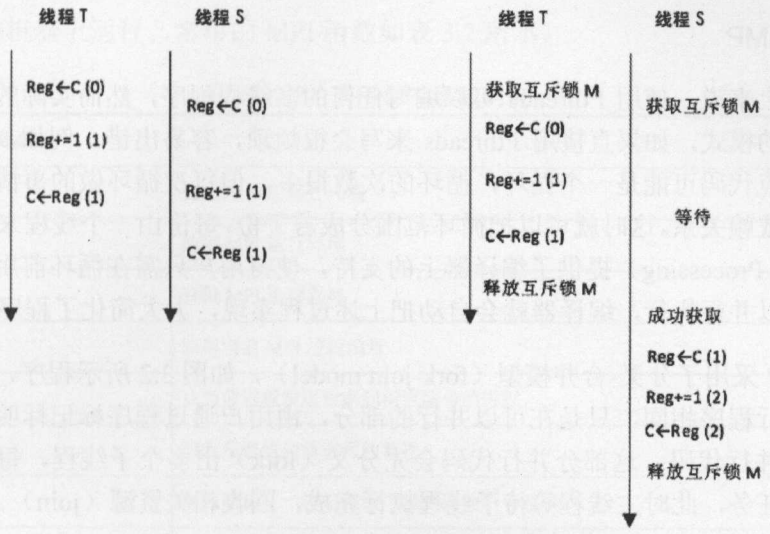


图 3.1 多线程竞争场景与互斥锁的使用

为了避免错误的发生，可以给资源加上互斥锁（mutex，即 mutual exclusion 的简称）。当一个线程 T 访问一项资源 R 时，先获取资源 R 的互斥锁 M，再对 R 进行操作，操作结束后释放互斥锁 M。如果线程 T 在操作 R 的同时，有另外一个线程 S 也要操作该资源，则线程 S 在申请互斥锁 M 时就会失败，一般会要求 S 等待直到线程 T 释放互斥锁 M，然后线程 S 就可以获取互斥锁 M，并对 R 进行操作。图中右边的情况就是在使用互斥锁保护计数器 C 之后的情况，无论线程之间的执行顺序如何，结果一定是正确的。从图中也可以看出，引入互斥锁后被保护的代码段就无法并行执行，会降低程序的效率。

当然，如果操作 O 本身是满足原子性（atomic）的，也就是对于系统来说，要么看到的是操作 O 发生之前的状态，要么是操作 O 发生之后的状态，那么就不需要互斥锁的保护了。上面例子中提到的计数器增一操作在 x86 体系结构上有相应的原子指令（LOCK ADD/XADD），如果使用这样的指令，硬件就可以保证操作的原子性，无须互斥锁的保护。一般来说，互斥锁会带来一些性能上的额外开销，尤其是当很多个线程同时尝试获取同一个互斥锁的时候。相比而言，这些硬件保证的原子操作性能就会好很多。因此，合理使用免锁（lock-free）的数据结构是提高多线程程序性能的有效途径。

当两个线程互相等待对方释放各自获取的锁时，就会发生死锁（deadlock）现象。这时，两个线程都不能继续执行，程序就会卡住而无法正常工作。当然，实际情况可能会更加复杂，可以是多个线程依次等待构成一个环，而且死锁不光会发生在多线程情况下，也有可能会发生在分布式情况下。死锁是复杂的并行程序容易发生的逻辑错误，而且调试起来也比较麻烦，需要通过程序运行的日志找出发生死锁的位置，进而分析发生死锁的原因。

2. OpenMP

从理论上来说，使用 Pthreads 可以编写任何的多线程程序，然而实际的应用往往会遇到一些并行的模式，如果直接用 Pthreads 来写会很烦琐，容易出错。例如，很多计算相关的程序的热点代码可能是一个循环，循环的次数很多，但每次循环做的事情是独立的，没有数据上的依赖关系。这时就可以把循环范围分成若干份，每份由一个线程来执行。OpenMP（Open Multi-Processing）提供了编译器上的支持，使得用户只需在循环前用程序标记标注这个循环可以并行执行，编译器就会自动把上述过程实现，大大简化了程序。

OpenMP 采用了分叉-合并模型（fork-join model），如图 3.2 所示程序，从逻辑上看与单线程的串行程序相同，只是在可以并行的部分，由用户通过程序标记标明，编译器自动将其转化为并行代码。这部分并行代码会先分叉（fork）出多个子线程，每个子线程执行一部分工作任务，此时主线程等待子线程执行完成，回收相关资源（join）。

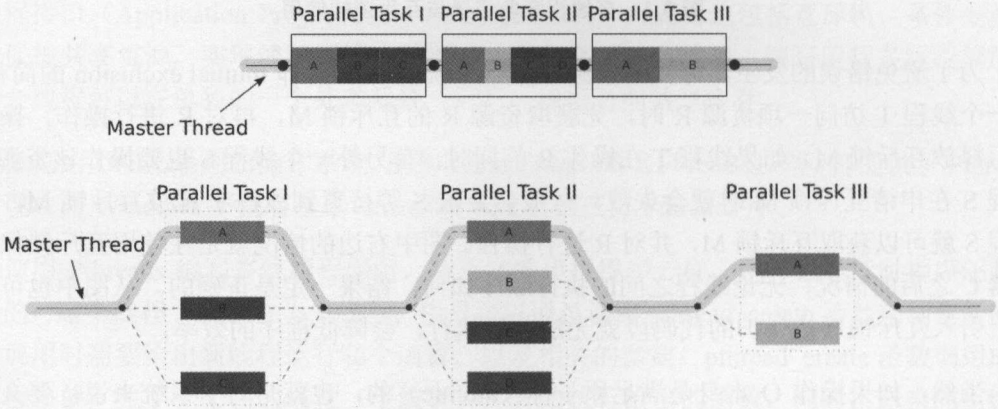


图 3.2 分叉-合并模型，来源：Wikipedia

3. MPI

上面介绍的 Pthreads 和 OpenMP 提供了单机多核环境下并行编程的工具，对于多机环境，由于节点之间不共享内存，一般都采用消息传递（message passing）的编程模型。MPI（Message Passing Interface）是目前应用最广泛的消息传递编程框架，它提供了多机之间通信、同步、管理等功能，通信又分为点对点通信（Send、Recv 等）和集体通信（Bcast、Scatter、Gather、Reduce 等），使用起来非常灵活。此外，新版本的 MPI 标准还包括并行 I/O、单边通信等高级功能。

在 MPI 中，程序在运行时会同时存在多个实例，被称为进程，进程之间不共享内存，

可以在不同的机器上运行。常用的 MPI 函数如表 3.2 所示。

表 3.2 常用 MPI 函数列表

函数名	作用
MPI_Init	初始化 MPI 运行环境
MPI_Finalize	结束 MPI 运行环境
MPI_Comm_size	获得 MPI 进程总数
MPI_Comm_rank	获得当前 MPI 进程编号
MPI_Send	从当前进程发送数据到指定编号的进程
MPI_Recv	接收发送给当前进程的数据
MPI_Bcast	将数据从某一特定进程广播到其他所有进程
MPI_Barrier	等待所有进程运行到该函数，然后继续执行
MPI_Scatter	将某一进程的数据分块发送给所有进程
MPI_Gather	从所有进程收集数据，拼接保存在某一进程中
MPI_Reduce	从所有进程收集数据，用某种运算聚合出结果

MPI 程序可以是单程序多数据（Single Program Multiple Data，SPMD），也可以是多程序多数据（Multiple Programs Multiple Data，MPMD），也就是说，在多个节点上，可以运行同一份代码，也可以运行不同的代码，唯一的要求就是它们之间能够协同工作。 MPI 的应用场景是超级计算机，其组成节点的可靠性一般都比较高，因而在容错方面考虑较少。目前 MPI 的容错一般采用检查点（checkpoint）的方式，也就是在同一时刻让所有的进程都把自己的状态存入磁盘，这样如果后续发生硬件方面的错误，可以在错误修复后将最近的一个检查点载入内存，减少重复计算的工作。目前常见的 MPI 实现有 MPICH、OpenMPI、Intel MPI 等。

4. 其他支撑软件

除了计算框架外，超级计算机还需要任务调度软件，用于管理节点，接受用户提交的计算任务，并将计算任务分配到可用的计算节点上运行。同时，调度软件还要负责控制用户的权限和配额、记账等功能。常见的任务调度软件有 SLURM、OpenPBS 等。 另外，超级计算机在执行任务时会遇到高并发的 I/O 操作，需要并行文件系统的支持。常见的并行文件系统有 PVFS、Lustre 等，它们可以安装在多个存储节点上，向计算节点提供全局一致的名字空间，用于存储计算任务的输入、输出文件和中间结果。

3.3 虚拟化和云计算技术

云计算是信息技术行业最重要的技术之一。什么是云计算？有很多种不同的解释。美国国家标准技术研究所（National Institute of Standards and Technology，NIST）认为，云计算是通过网络使得一组可配置的计算资源（例如网络、计算机、存储、应用程序、服务等）能够在任何地点、方便地、按需地进行访问的模型，资源的提供和释放可以快速完成，管理开销低，与提供商的交互简便易行。

云计算有3种服务模型，分别是软件即服务（Software as a Service，SaaS）、平台即服务（Platform as a Service，PaaS）和基础架构即服务（Infrastructure as a Service，IaaS），向消费者提供不同层次的服务。其中SaaS向消费者提供具体的应用软件服务，PaaS则让消费者可以通过一定的API开发自己的应用，部署在提供商的环境中，而IaaS则直接向消费者提供虚拟机、存储空间等计算资源，它们的关系如图3.3所示。云计算的部署模型根据受众的不同可分为私有云、社区云、公有云和混合云。云计算的出现使得大数据处理的门槛降低，普通的开发者和用户可以用较为经济的成本获得处理大数据的能力。

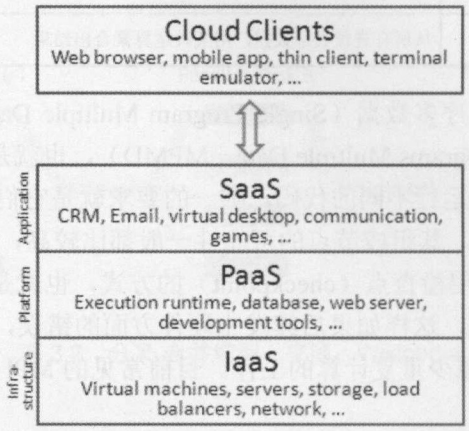


图3.3 云计算的服务模型，来源：http://en.wikipedia.org/wiki/Cloud_computing

3.3.1 虚拟化技术

虚拟化技术是指创建虚拟的事物，包括计算机硬件平台、操作系统、存储设备、计算机网络等，是云计算的支撑技术。这些虚拟的事物具有和真实的、物理上存在的事物相同的外部表现，例如虚拟机可以像真实的计算机一样安装操作系统、运行应用程序，虚拟存

储设备可以像真实的硬盘一样保存数据。早在 20 世纪 60 年代，当时的大型计算机比较稀少，就产生了虚拟化技术，能够将一台大型计算机的资源在逻辑上划分为多台虚拟机，分别运行不同的应用程序，提高资源的利用效率。现代的硬件虚拟化技术早期完全由软件来实现，虚拟机在遇到系统相关的特权指令时要通过特殊的指令转换过程来处理。后来 Intel 和 AMD 都推出了支持虚拟化的专用指令，使得这些工作可以由处理器直接完成，提升了虚拟机的执行效率。虚拟化技术可以把物理硬件资源的一部分抽取出来并封装成逻辑上独立的虚拟机，来满足客户不同的需求，实现云计算所要求的资源灵活配置。

传统的硬件虚拟化会在物理机（或主机、宿主，Host）上运行虚拟机管理器（Hypervisor，或 Virtual machine manager），负责虚拟机的创建、调度和管理。虚拟机（或客户机，Guest）创建之后，需要像实际的机器一样安装操作系统，或者使用已经准备好的系统镜像。由于虚拟化的是一台完整的机器，因此虚拟机上安装的系统只受物理硬件架构的限制，而与主机操作系统无关。例如在 Linux 的机器上创建运行 Windows 的虚拟机，或者在 Windows 的机器上创建运行 Linux 的虚拟机，如图 3.4 左边所示。常见的商用硬件虚拟化软件有 VMware 等，开源的有 Xen、KVM、VirtualBox 等，其中 Xen 和 KVM 侧重服务器领域的虚拟化，而 VirtualBox 主要是桌面领域的虚拟化。

硬件虚拟化技术中主机和客户机运行不同的操作系统，提供了非常好的灵活性，然而对于很多应用场景，这样的灵活性用处不大，客户机的操作系统需要占用独立的硬件资源，反而带来了额外的开销。操作系统虚拟化可以更好地应对这样的应用场景。在操作系统虚拟化中，客户机（也被称为容器，Container）和主机共享同一个操作系统，在操作系统内把所需的资源封装成容器，用于运行应用程序，如图 3.4 右边所示。常用的操作系统虚拟化软件有 LXC 等，而 Docker 则是在此基础上又进行了一次封装，使得能够正常运行特定应用程序的环境，包括系统库、语言运行时、第三方库等。它以容器镜像（Container image）的形式发布，供用户直接下载使用，很好地解决了现在大型软件依赖关系复杂、配置烦琐的问题。

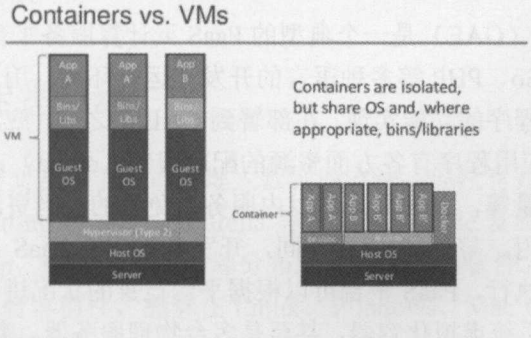


图 3.4 虚拟机和容器，来源：http://www.rightscale.com/

3.3.2 云计算服务

云计算服务大大降低了一般开发者和用户的时间、经济成本。在云计算服务出现以前，用户架设网站需要自己购买服务器，并联系有良好电气和网络条件的机房。把服务器放入机房后，如果出现远程无法解决的问题，还得亲自去机房现场。服务器的购置费和机房的托管费都是一笔不小的开销。有了云计算服务之后，用户通过直观的 Web 界面或简洁的 API 就能创建和使用虚拟机，根据应用的负载用户可以随时调整虚拟机的配置。提供商按照资源的使用量和使用时间收取服务费。

常见的 IaaS 云计算服务有亚马逊的 AWS（Amazon Web Services）、微软的 Azure、阿里巴巴的阿里云等。以 AWS 为例，它提供了计算、网络、存储等多种不同的服务，配合起来可以构成应用所需的运行环境。AWS 提供的服务主要包括如下。

- Amazon Elastic Compute Cloud（EC2）：虚拟云主机服务。
- Amazon Simple Storage Service（S3）：基于 Web 服务的存储。
- Amazon Elastic Block Store（EBS）：为 EC2 提供的持久化的块存储。
- Amazon DynamoDB：可扩展、低延迟的 NoSQL 数据库服务。
- Amazon Relational Database Service（RDS）：关系型数据库服务。
- Amazon Route 53：高可靠的域名系统（DNS）服务。
- Amazon CloudFront：内容分发网络（CDN）服务。
- Amazon Elastic MapReduce（EMR）：在 EC2 和 S3 的基础上用 Hadoop 搭建的 MapReduce 服务。

其中，最为核心的虚拟机服务 EC2 有多种不同的虚拟机类型，根据常见的应用需求有不同的处理器核数、内存大小，对科学计算任务还专门配置了 GPU 的类型。其他 IaaS 云计算服务提供商也提供了与 AWS 类似的服务。

Google App Engine（GAE）是一个典型的 PaaS 云计算服务平台。GAE 给开发者提供了包括 Python、Java、Go、PHP 等多种语言的开发和运行环境，用于编写 Web 应用程序。开发者只需专注于应用程序的功能实现，在部署到 GAE 上之后，应用程序的性能将由平台来保证。在 GAE 上的应用程序有各方面资源的配额限制（quota），例如每天的 HTTP 请求数量、数据库操作数量等，在配额限制以内服务是免费的。当资源超过配额限制时，会根据规则收取一定的费用。与 IaaS 服务不同，开发者在使用 PaaS 服务时并不知道自己的应用程序在哪台机器上执行，PaaS 平台可以根据平台自身的状况进行调度，可能是一台虚拟机，或者是一个操作系统虚拟化容器，甚至是多台物理服务器。类似的 PaaS 服务还有新浪 App Engine、Red Hat 的 OpenShift 等。

使用云计算服务的另外一大好处是服务提供商已经对平台采取了必要的安全措施，可以在很大程度上避免恶意攻击和系统漏洞对应用产生的破坏。

3.4 基于分布式计算的大数据系统

单台计算机的能力是有限的，而需要处理的问题规模在不断地增长。为此，人们开始探索用多台计算机组成一个系统进行协同处理。这样的多机系统复杂程度要远远高于单机系统，会遇到很多新的问题。首先，数据要分布在不同的机器上，在其他机器上的数据无法通过本地的内存访存或磁盘操作来获得，必须进行网络通信。由于网络的带宽是有限的，因此当机器数量多到一定程度时通信有可能会成为瓶颈。此时，系统的性能不再因添加新的机器而提升，可扩展性会成为问题。其次，尽管单台机器出故障的概率不高，然而当机器数量多到一定程度时，其中一台机器出现故障的概率是很高的，当部分机器出现故障时，我们希望多机系统作为一个整体还能够正常工作，这是可靠性的问题。再次，在这样一个多机系统中，用户编写的程序会在不同的机器上同时运行，运行环境比单机的情况要复杂很多，如何降低用户的使用门槛，使得用户能方便地进行数据处理，给多机系统的编程模型提出了挑战，这是易用性的问题。

这样的多机系统一般被称为分布式系统。为了解决分布式系统的各种问题，人们早在20世纪70年代就开始研究分布式系统的理论，然而大规模的实践是由Google公司在2000年左右开始的。为了方便地处理互联网产生的海量数据，同时控制成本，Google采用了大量普通机器通过网络连接并由专门的软件系统控制协同工作的解决方案，先后发表了“Google File System（Ghemawat, et al. 2003）”、MapReduce（Dean & Ghemawat 2008）、Bigtable（Chang, et al. 2008）等论文，介绍了分布式文件系统、数据处理系统、半结构化存储系统等的设计和实现。

3.4.1 Hadoop生态系统

Google虽然通过论文发布了公司处理大数据的方法，但没有开源他们的软件系统。从2005年开始，Doug Cutting和Mike Cafarella等人根据Google发表的论文和他们自己的实践经验，开发了Hadoop，成为主流的开源分布式大数据处理软件。Hadoop主要用Java语言编写，具有较好的平台移植性，能够在Linux、Windows、Mac OS X等各种常见的操作系统下运行。经过10年的发展，Hadoop已经成为Apache开源软件基金会旗下的重要项目，

并演化出了一个较为完整的生态系统。

Hadoop 具有很好的可扩展性，2012 年 6 月，Facebook 宣布他们部署了当时最大的 Hadoop 集群，HDFS 的容量超过 100PB。2013 年，Yahoo!在一个由 42,000 台机器组成的集群上部署了 Hadoop 系统，每天运行大约 500,000 个 MapReduce 任务。

在 Hadoop 系统中，每种服务一般都有若干种角色，在分布式环境中负责不同的功能。每种角色的实例都是一个进程，不同的进程可以在同一台机器上运行，也可以根据配置运行在不同的机器上。

1. HDFS

HDFS（Hadoop Distributed File System）是 Hadoop 的分布式文件系统。和 GFS 一样，HDFS 将文件按一定的大小切块（默认设置是 128MB），然后把每个块以多个副本的形式保存在不同的数据节点（Data Node）上。通常每个块会保存 3 个副本，多个副本可以保证在少量节点出现问题时数据不会丢失，同时也能提高数据读取的速度。

在 HDFS 中，文件的元数据，包括文件路径、大小、创建日期、所有者，以及分块情况和每块所在的数据节点编号等，会保存在名字节点（Name Node）上，名字节点全局只有一台，同时还可以配置一个次要名字节点（Secondary Name Node）来同步这些元数据。一旦名字节点出现故障，就可以很快地从次要名字节点恢复所有的元数据。

用户从客户端（Client）访问 HDFS 时，会先和名字节点通信，获得所要操作的块所在的数据节点编号，然后再和数据节点通信，完成数据的读/写操作，读/写过程如图 3.5 所示。

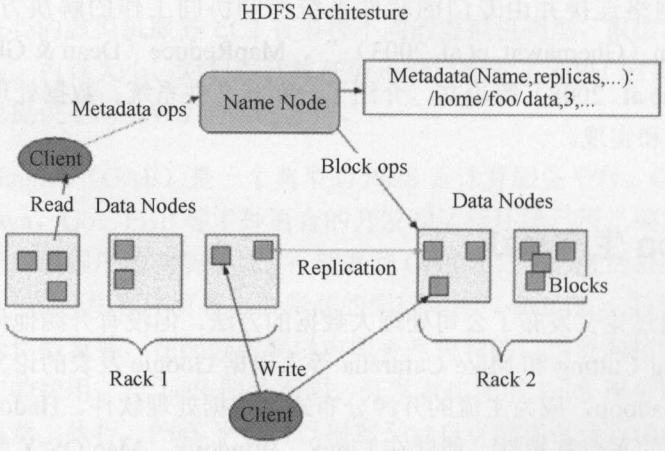


图 3.5 HDFS 的读/写过程，来源：http://hadoop.apache.org/

HDFS 的设计目标是存储那些一次写入多次读取的大量数据，例如搜索引擎会用爬虫抓取大量的 Web 页面，一旦存入分布式文件系统之后，这些数据不会再被修改，但可以追加新的数据。对于桌面用户共享文件的应用场景（有很多小文件，内容会经常发生修改），HDFS 一般是不适用的。

2. YARN 和 MapReduce

YARN（Yet Another Resource Negotiator）是 Hadoop 的计算资源管理和调度系统，接受任务请求，并根据请求的需要来分配资源，调度任务的执行。在 YARN 中，有一个资源管理器（Resource Manager）节点，负责全局的资源分配；每个计算节点的资源由节点管理器（Node Manager）控制。当客户端（Client）提交任务时，YARN 会在某个计算节点上创建一个应用程序主控器（Application Master）来控制整个任务的执行流程，同时根据需要在一些计算节点上创建容器（Container），包含一定数量的处理器和内存，用于该任务的执行，应用的执行流程如图 3.6 所示。

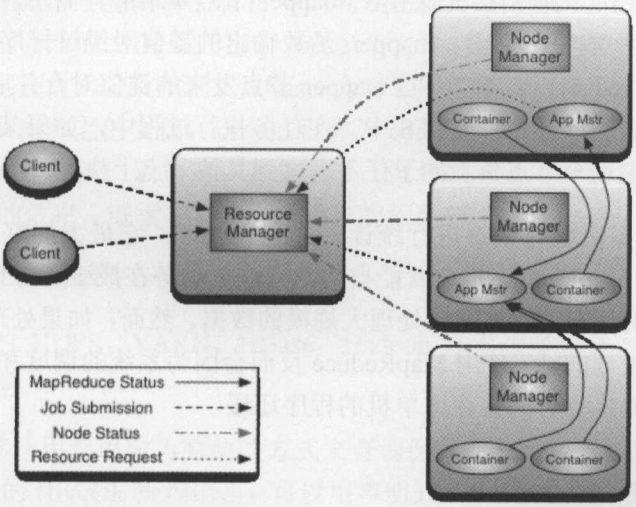


图 3.6 YARN 调度执行应用程序的流程，来源：http://hadoop.apache.org/

YARN 是 Hadoop 的第二代计算资源管理和调度系统，第一代系统只能执行 MapReduce 任务，而 YARN 的设计更加通用，除了可以执行 MapReduce 外，还可以执行 MPI 等传统的并行程序（https://github.com/alibaba/mpich2-yarn）。

MapReduce 是 Google 提出的并行程序编程模型。它把任务的处理流程分为 map 和 reduce 两个阶段，每个阶段用户可以编写一段串行代码来处理数据：在 map 阶段，输入数据被分割成一些基本的项（例如，每行文本或者每个单词），用户编写的 mapper 函数逐项

接受输入，经过自定义的处理后可以发射一些键值对（key-value pair）；在 map 和 reduce 阶段之间，map 阶段发射的这些键值对通过排序，具有相同键的键值对被整理到一起；在 reduce 阶段，用户编写的 reducer 函数接受整理后的键值对，经过自定义的处理后可以发射新的键值对。reduce 阶段发射的键值对就是 MapReduce 任务最终的输出结果。

以统计文档中不同单词的出现次数为例，不考虑单词的各种变形和标点符号等因素，在 map 阶段把文档按照空白字符分割成单词，作为 mapper 函数的输入。mapper 函数每遇到一个单词 w，就输出一个（w, 1）的键值对，表示发现 w 在这里出现了 1 次。reducer 函数则把同一个单词 w 的所有键值对的值相加，设最终结果为 f，输出键值对（w, f），表示单词 w 一共出现了 f 次。

在执行时，MapReduce 框架会根据具体情况分配一定数量的 mapper 和 reducer 节点，用于执行 mapper 和 reducer 函数。输入在任务执行前已经保存在 HDFS 中，分配 mapper 节点时会根据就近的原则尽量分配到包含有该输入数据块的数据节点上，减少网络通信开销。mapper 函数输出的键值对会先保存在 mapper 节点本地的存储中，并进行排序。每个 reducer 节点会负责一段范围的键，mapper 函数输出的键值对经过排序后会转发给特定的 reducer 节点，每个 reducer 节点将不同 mapper 节点发来的键值对合并后交给 reducer 函数进行处理，最终的结果将保存在 HDFS 中。在任务执行过程中，如果某个计算节点发生故障，YARN 会自动把这个节点负责的子任务调度到其他节点上执行。

MapReduce 模型适合对数据进行统计、分类等处理，它最大的好处在于当用户实现 MapReduce 任务后，MapReduce 框架就能自动地把这个任务在成千上万台机器上调度执行，同时处理机器故障的情况，非常适合处理大规模的数据。然而，如果处理的数据规模较小，一台机器就能很快处理的话，使用 MapReduce 反而会因为系统的调度和多机之间的通信而带来额外的开销，其执行时间往往比单机的程序还慢。

3. HBase

HBase 实现了 Bigtable 论文提出的基于列的分布式存储。在 HBase 中，数据以表（Table）的形式组织，每个表可以有很多行（Row），每行可以有若干个列族（Column family），而每个列族可以包含多个列（Column）。列族需要事先定义，而列可以在使用中随时添加，无需事先定义。每行每列会对应一个单元（Cell），而一个单元的值可以有多个版本，用不同的时间戳来区分（如图 3.7 所示）。每个值是一个任意长度的字节串，因此可以用来保存任何类型的数据。每行都有一个用户自定义的主键，作为引用该行的 id。

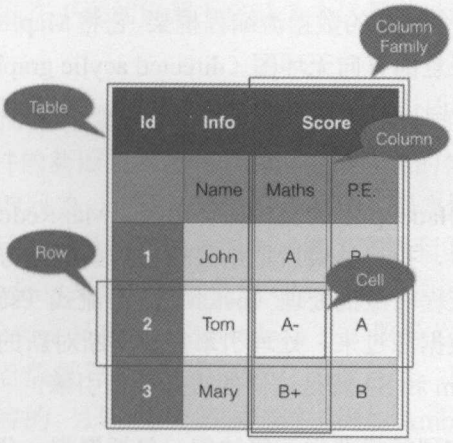


图 3.7 HBase 的数据组织形式

HBase 中的数据可以用 id 和列来随机访问，也可以顺序地访问一段连续 id 的行。与关系型数据库不同，HBase 不能做基于非 id 列的随机访问。 HBase 的实现以 HDFS 为基础，每个表的每个列族都会对应 HDFS 上的一个或多个文件。新插入的数据都会追加到文件的末尾。当失效的数据超过一定比例时，HBase 会对这些文件进行重写，把有效的数据依次写入到新的文件，并删除老的文件，这个过程称为 compact 操作。 HBase 可作为数据仓库存储有一定结构的海量数据，数据可以发生修改，但最好不要十分频繁。频繁的修改会降低 HBase 读取操作的效率，或增加 compact 操作的频率的工作量。

4. Hadoop 其他常用组件

除了上面介绍的 HDFS、YARN、HBase，Hadoop 还有很多组件，用于解决大数据处理中各个方面的问题。

Hive 和 Pig 能够让用户用较为简便的方式来查询保存在 HDFS 或 HBase 中的数据。Hive 提供了类似于 SQL 的 HiveQL 查询语言，可以用声明式（declarative）编程的模式来查询数据。Pig 则提供了 Pig Latin 脚本语言，可以用命令式（imperative）编程的模式来查询数据。无论是 Hive 还是 Pig，查询最终都会转换成 MapReduce 任务来执行，但是大大减少了手工编写 MapReduce 任务的工作量，减小了出错的机会。Hive 和 Pig 对从事数据分析的用户而言是非常有用的工具。

ZooKeeper 提供了编写分布式软件所需的常用工具，包括分布式系统的名字服务、配置管理、同步、领导者选举、消息队列、通知系统等。很多 Hadoop 组件，例如 HBase，就使用了 ZooKeeper 提供的这些功能。如果有需要，开发者也可以直接调用这些功能，编写的分布式软件可以运行在 Hadoop 系统上。

Tez是比MapReduce更一般化的数据流编程框架，它将MapReduce规定的map和reduce两阶段的执行流程推广为任意的有向无环图（directed acylic graph，DAG），用图中的每个顶点表示一个处理步骤，有向边表示数据的流向。目前，Hive、Pig等组建的执行引擎正在从MapReduce向Tez过渡。

Storm和S4是建立在Hadoop上的流式处理引擎。MapReduce任务的输入会在执行前存放在文件系统或数据库中，执行结束后把结果输出到新的地方，任务执行过程中数据不会发生变化，这样的处理流程叫做批处理（batch）。而流式（streaming）处理指的是待处理的数据会源源不断地从数据源过来，处理引擎需要不断对新的数据进行处理，并随时输出具有时效性的结果。Storm和S4就是这样的流式处理引擎。

Mahout是用Hadoop实现的机器学习算法库，包括聚类、分类、推荐，以及线性代数中的一些常用的算法。用户可以直接调用Mahout提供的算法，而不必自己再用MapReduce实现一遍。老版本的Mahout中的算法主要用MapReduce来实现，新版本中的算法使用一种支持线性代数操作的领域特定语言（Domain Specific Language，DSL）来实现，用这种DSL实现的算法可以很容易地在Spark平台（见下文）上自动优化和并行执行。有意思的是，Mahout在印度语中是骑象人的意思，而Hadoop则是作者Doug Cutting孩子的玩具象的名字。

Giraph在Hadoop上实现了类似于Google的Pregel这样的图计算引擎，用于处理Web链接关系图、社交网络等类型的数据。关于图计算的内容会在下文详细讨论。

Sqoop是一个命令行工具，用于在Hadoop和传统的关系型数据库之间传输数据。通过它可以增量式地把数据从MySQL等数据库的表格导入到HDFS或HBase中，也可以反过来把数据从Hadoop系统导出到数据库，实现了Hadoop系统与传统软件的数据交换。

日志的收集和分析是Hadoop平台的一个重要应用。在Hadoop生态系统中，Chukwa、Flume、Kafka、Scribe都能进行日志的收集，收集的结果会导入到HDFS。这些软件由不同的公司主导开发，各有特点，可以根据需要选取使用。它们之间的比较可参见http://dongxicheng.org/search-engine/log-systems/。

5. Hadoop的安装部署和开发

Hadoop系统有3种安装模式，分别为单机模式、伪分布式模式和全分布式模式。在单机模式下，服务一般只启动一个进程，提供和分布式环境相同的接口，可用于Hadoop应用程序开发和正确性调试。伪分布式模式在同一台机器上启动多个进程，代表服务的不同角色，这些进程之间的通信在本地完成，不通过网络，一般用于调试系统在分布式配置下

的正确性。全分布式模式则在不同的机器上启动多个进程，进程之间的通信要通过网络，一般用于产品环境的部署。

Hadoop 生态系统中的组件数量很多，每个组件都是由不同的开发团队负责，版本更新的频率也都不同，不同组件的新旧版本之间容易出现不兼容的情况，Hadoop 系统和操作系统之间也可能出现不匹配的问题。对于一般的用户而言，想要快速获得可用的 Hadoop 环境，一是可以直接使用云计算提供商提供的服务，像上文提到的 AWS 的 EMR 等。如果要在自己本地的集群部署 Hadoop 系统，推荐使用 Cloudera 公司的 CDH 或 Hortonworks 公司的 HDP 发行版。这两个公司是目前最大的 Hadoop 技术服务公司，同时也是 Hadoop 社区最大的贡献者。它们的发行版推荐了相适应的操作系统环境，同一个发行版内的组件之间经过严格的测试是可以兼容的。另外，发行版也提供了例如 Ambari 这样的安装和管理工具，能通过 Web 界面方便地安装部署 Hadoop 系统，避免了烦琐的手工配置步骤。

在 Hadoop 系统上开发应用时，由于依赖的 Java 库较多，同时也存在版本的问题，推荐开发者使用 Maven 工具来管理项目。Maven 可以添加指定版本的库，在编译时会自动从网上下载所有依赖的库，避免配置带来的麻烦。

3.4.2 Spark

MapReduce 给用户提供了简单的编程接口，用户只需要按照接口编写串行版本的代码，Hadoop 框架会自动把程序运行到很多机器组成的集群上，并能处理某些机器在运行过程中出现故障的情况。然而，在 MapReduce 程序运行过程中，中间结果会写入磁盘，而且很多应用需要多个 MapReduce 任务来完成，任务之间的数据也要通过磁盘来交换，没有充分利用机器的内存。为此，美国加州大学伯克利分校的 AMPLab 设计实现了 Spark 计算框架（Zaharia, et al. 2012），充分利用现在机器的大内存资源，使得大数据计算的性能得到了进一步的提升。Spark 由 Scala 语言编写，Scala 是一种基于 Java 虚拟机的函数式编程语言，因此 Spark 提供的操作和 MapReduce 相比更加丰富和灵活。

Spark 设计的核心是一种叫做可靠分布式数据集（Resilient Distributed Dataset，RDD）的数据结构。一个 RDD 是一组数据项的集合，可以是普通的列表，也可以是由键值对构成的字典。在 Spark 中，一个 RDD 可以分布式的保存在多台机器上，可以保存在磁盘上，也可以保存在内存中。对 RDD 的操作分为动作（action）和变换（transformation）。表 3.4 列出了 RDD 支持的常见操作。与 MapReduce 不同，Spark 的操作都是对 RDD 整体进行的，而不是对具体的每一个数据项。动作操作会直接生效，产生新的 RDD，而变换操作的执行则是懒惰的（lazy），操作会被记录下来，直到遇到下一个动作时才产生一个完整的执行

计划。Spark 中的 RDD 可以由框架自动或由开发者人为地指定缓存在内存中，在内存足够的情况下对于某些应用可以获得比 MapReduce 快 100 倍以上的性能。

表 3.4 RDD 支持的常用基本操作

类型	操作	作用
变换	map	给出一个元素映射的函数，把一个 RDD 映射为另一个 RDD，新的 RDD 中的元素是老的 RDD 中每个元素的映射结果，类似于 MapReduce 中的 map
	filter	对一个 RDD 中的元素进行过滤，得到新的 RDD
	sample	确定性采样
	reduceByKey	把键相同的元素所对应的值用一个函数聚合起来，类似于 MapReduce 中的 reduce
	union	合并两个 RDD
	join	把两个 RDD 按照键进行连接，类似于 SQL 的 JOIN 操作
	sort	排序
	partitionBy	用一个划分函数对 RDD 中的元素进行划分
动作	count	统计 RDD 中的元素个数
	collect	将 RDD 中的元素导出为一个序列
	reduce	将 RDD 中的元素聚合为一个值
	save	将 RDD 输出到 HDFS 等存储系统中

Spark 可以独立运行，也可以在 Hadoop 系统上运行，由 YARN 来调度。Spark 支持对 HDFS 的读/写，因此 MapReduce 程序可以很容易地改写成 Spark 程序，并在相同的环境下运行。

与 Hadoop 类似，Spark 也提供了一些组件，用于不同的应用场景。前面介绍的 Spark 核心组件被称为 Spark Core。Spark SQL 在 Spark Core 的基础上提供了新的数据抽象 SchemaRDD，用于处理结构化和半结构化的数据，支持用 SQL 的语法对 SchemaRDD 进行查询。与 Hive 类似，Spark Streaming 提供了流式处理的功能，与 Hadoop 的 Storm/S4 类似。MLlib 是 Spark 上的机器学习算法库，提供了类似 Mahout 的功能。而 GraphX 则是 Spark 的图计算扩展框架，能够完成与 Giraph 相似的功能。

总地来说，目前 Spark 已经发展到比较成熟的阶段，其核心功能涵盖了 Hadoop 的大部分内容，并且可以在 Hadoop 生态系统内使用，具有性能上的优势，正在获得越来越广泛的应用。

3.4.3 典型的大数据基础架构

对于海量的数据，各个互联网公司都搭建了自己的大数据基础架构，并通过论文或者开源软件的形式公开了一些组件。

这其中最有代表性的是 Google 公司。在 Google 公司的基础架构中，处于最底层的是 Google 文件系统（Google File System, GFS）。GFS 面向搜索引擎设计，文件的内容可以不断追加，但是不能修改。GFS 是分布式的，描述文件的信息（又称为元数据）保存在文件系统的主节点上，另有多个数据节点，负责保存分块之后的文件内容。同一个数据块分别保存在多个数据节点上，即使出现一定数量的数据节点故障，整个文件系统依然可以正常工作，不会导致数据丢失。在计算方面，Google 提出了 MapReduce 的计算模式，并可以通过 Sawzall 语言来进行类结构化的查询和计算。此外，对于半结构化数据，Google 用 Bigtable 来存储和查询，而 Chubby 则提供了一般分布式系统所需的同步等基本服务。Spanner 则提供了跨地域的分布式结构化数据库服务。

微软公司也在内部搭建了一套大数据基础架构，包括分布式文件系统 Cosmos、分布式计算系统 SCOPE 等，并可以通过 LINQ 的方式用.NET 平台上的语言来开发应用程序。这些系统为搜索引擎 Bing 提供了支撑。

Yahoo 和 Facebook 公司主要使用 Hadoop 来构建大数据系统，并在此基础上开发了 Prism、Corona 等新的组件，能够在一个公司内部更好地管理集群，并提供更方便的数据查询和分析功能。

阿里巴巴开源了一系列大数据系统软件，包括分布式文件系统 TFS、分布式键值存储系统 tair、分布式数据库 OceanBase 等。与 GFS、HDFS 等为大文件设计不同，TFS 主要为小文件设计（例如，淘宝上的商品图片），能够有效地支撑淘宝的 CDN 服务。

3.5 大规模图计算

图（graph）是一种重要的数据抽象模型，由顶点（vertex）和边（edge）构成。顶点可以表示对象，而边则表示对象之间的关系。例如，在互联网的拓扑结构图中，顶点代表网络中的设备，可以是交换器、路由器，也可以是使用网络的计算机，边则代表它们之间的链接。社交网络中，顶点代表用户，边则表示用户之间的好友关系，或互动的动作（如微博的@）。

图的分析可分为图的查询和计算两大类。图的查询是指在图中查找符合一定条件的顶点、路径或子图，这类问题可以由图数据库来解决，比如上文提到的 Neo4j。图的计算则是指根据图的拓扑结构以及顶点和边上所带的属性经处理得出所需结果的过程，以图的整体为输入的算法都属于图计算的范畴，例如广度优先搜索（breadth-first search，BFS）、深度优先搜索（depth-first search，DFS）、连通分支、PageRank 等。当图的规模不太大时，图计算的算法可以手工从头编写，也可以调用一些现成的软件库，如 Boost Graph Library、SNAP、NetworkX 等。

当图的规模变大时，图计算会面临挑战，主要原因都是由图的不规则性引起的。由于图中的边可以连接任意的两个顶点，而图的算法在访问一个顶点时经常要访问和它相关的边以及邻居顶点，这些数据在计算机存储上的局部性就会很差。局部性差会使计算机的缓存机制效果变差，从而导致访问速度变慢。在并行计算时，不规则性会导致图的顶点和边划分不均，使得参与计算的多台机器的负载不平衡，影响系统整体的效率。另外，由于可能存在很多跨机器的边，计算过程中需要进行大量的通信操作，也会影响计算的速度。

尽管像 MapReduce 这样的通用计算模型也能实现一部分的图算法，从而使这些算法可以在分布式环境中执行，但是图计算还是有它自己非常明显的特点，这是因为使用通用的计算模型一来编写代码比较烦琐，二来执行的效率也不高。下面将介绍一些专用的图计算框架。

3.5.1 分布式图计算框架

Google 在 2010 年发表论文提出了在分布式环境下进行大规模图计算的框架 Pregel（Malewicz, et al. 2010），随后 Hadoop 也根据 Pregel 的原理实现了开源的 Giraph 组件。这些框架提出了类似 MapReduce 的编程模型，简化了开发者的工作，同时使得编写的程序能够在分布式环境下高效、容错地执行。

Pregel 借鉴了 BSP（bulk synchronous parallel）模型，采用以顶点为中心（vertex-centric）的编程方式。整个计算过程被分成若干步（superstep），每步计算中，每个活跃的顶点都会从指向它的边收到上一步从邻居顶点发来的消息，根据这些消息以及顶点本身的状态可以计算出一些新的结果，并通过从它发出的边以消息的形式传递给其他邻居顶点，这些邻居顶点将会在下一步开始时收到发给自己的消息。每一步计算结束时，顶点可以选择将自己的状态改为非活跃，表示至此该顶点认为自己的计算过程已经结束，然而如果下一步开始时发现它收到了新的消息，它就会被重新激活，继续计算。当某步结束时如果所有的顶点都处于非活跃的状态，则整个图计算的过程结束。这时，所需的结果都保存在每个顶点的状态里。从用户编写代码的角度来看，这些计算和操作都发生在每个顶点上，因此被称

为以顶点为中心的编程方式。

我们知道在常见的社交网络等图中，顶点的度数呈幂律分布（power-law distribution），简单地说就是存在少数度数非常大的顶点，同时还有大量度数很小的顶点。这会导致按照完整的顶点作为基本单位进行图的划分时很难做到均衡。为了解决这个问题，GraphLab（Gonzalez, et al. 2012）对 Pregel 的计算模型进行了改进，允许把度数很大的顶点拆分成多个副本，每个副本只与该顶点的一部分边相连，这些副本可以被划分到不同的机器上，把划分的粒度从顶点一级降到了边一级，使负载更加均衡。顶点的计算被分成 Gather-Apply-Scatter 三个阶段，Gather 阶段从入边收集消息，Apply 阶段把收集到的消息中的数据进行聚合，然后修改顶点的状态，最后 Scatter 阶段以更新后的状态通过出边发送新的消息。如果一个顶点被分为多个副本，在 Apply 阶段时，每个副本各自完成聚合后要把所有副本的信息再聚合起来，得到该顶点实际的完整结果。改进后的计算框架与 Pregel 相比，在很多场景下有数量级的性能提升。

3.5.2 高效的单机图计算框架

当数据规模非常大时，我们可以用分布式系统来处理，期望以增加硬件资源的方式来提升性能。对于图计算而言，Pregel 和 GraphLab 就采取了这种思路。然而在实际情况中，图计算所面临的问题规模并不是无限大的，除了互联网网页的超链接关系的图规模很大外，一般社交网络所产生的图顶点个数最多也就是几亿到十几亿（与地球人口数量相当），边数一般是顶点的几十倍。这个规模的数据如果处理得当，单台计算机就能在合理的时间内完成计算任务。同时，尽管诸如 Hadoop、Spark、Giraph 这样的分布式计算框架大大降低了用户进行分布式编程和运行的复杂性，但和单机程序相比，在环境配置和调试上仍然要烦琐很多，需要一定的经验。而且也不是每个人都能拥有分布式的环境。GraphChi（Kyrola, et al. 2012）、X-Stream（Roy, et al. 2013）和 GridGraph（Zhu, et al. 2015）就是这样的单机图计算框架。

前面提到，图计算的处理之所以慢，主要的原因在于图的结构不规则，导致计算过程中随机访问较多，局部性差。单机系统进行计算时，由于内存无法容纳所有的顶点和边，因此还要进行磁盘读/写，局部性的影响会进一步放大。GraphChi 在计算前会对图的数据进行预处理，在磁盘上把边按照起点和终点排序，计算时将消息写入磁盘文件中，这样的顺序可以减少计算过程中访问的随机性。X-Stream 把计算过程中沿边传播的消息先顺序地添加到缓冲区中，此时消息是按照起始顶点有序的，然后通过类似于外部排序的方式把这些消息重新排列，使得它们按照终止顶点有序。这里有序并不是一种严格的升序或降序排列，而是一种比较宽松的顺序。在这样的设计下，消息的产生、重排和收集都可以用上磁盘的

顺序带宽，计算速度就很快。而 GridGraph 则把图的边按照起点和终点划分成二维的栅格，每次处理一个栅格，栅格中每条边的顺序并不重要，因此预处理时间较短。一个栅格对应的起点和终点的数据都可以同时存放在内存中，消息直接作用在顶点数据上，大大减少了所需的磁盘带宽，进一步提高了单机图计算的效率。

实验结果表明，对于有 4200 万个顶点和 15 亿条边的一个 Twitter 关系图，GraphLab 用 64 台机器、每台机器 8 核的集群进行计算需要 3.6 秒，而 GraphChi 在一台普通的台式机上用 158 秒也能完成同样的工作。同样是这个图，用 GraphChi 统计三角形个数需要 60 分钟，而用 Hadoop 系统在 1636 个节点上计算则花费了 423 分钟。X-Stream 在很多时候性能比 GraphChi 更好，而 GridGraph 的性能比前两者都要好。对于大多数情况来说，单机的图计算框架是一种更好的选择。

3.6 NoSQL

传统的关系型数据库以其规整的数据组织结构、方便的 SQL 查询语言以及严格的事务处理支持而获得广泛的应用，曾经是应用程序后端进行数据存储的唯一选择。然而随着互联网的飞速发展，Web 应用对数据规模和读/写性能的要求不断提高，各种 NoSQL 数据库如雨后春笋般地涌现，在特定的应用场景获得了比 SQL 更好的表现。

常见的 NoSQL 数据库大致可分为以下几类。

- 基于列的存储（column-oriented store）：数据组织成表格的形式，每个表格有行和列两个维度，一行表示一个数据项，而列的维度可以分为若干个层次（例如 HBase 中的列族和列），最小的层次被称为列。一行一列的交叉被称为一个单元，每个单元可以保存若干个版本的数据，用时间戳来区分。基于列的存储的例子有前面讲过的 HBase，以及 Cassandra 等。与关系型数据库不同，基于列的存储中对于一个特定的列并不是每行都有对应的单元。基于列的存储可扩展性较好，可用于海量数据的存储。
- 基于文档的存储（document-oriented store）：数据以文档为单位组织，一个文档可以包含若干属性，每个属性有各自的名字和值。不同的文档可以有不同的属性集合。例子有 MongoDB、CouchDB 等。基于文档的存储中数据的组织形式比较灵活，适合于需求变化快速的 Web 应用程序等场景。
- 键值对存储（key-value store）：数据以键值对的形式组织，操作非常简单（put、get、delete）。键值对存储又可细分为以下几类。

  - 单机磁盘型：例如 Berkley DB、LevelDB 等，数据持久化在磁盘上，强调单机读写性能，数据一般按照键排序，多用于本地应用的数据存储。
  - 单机内存型：例如 memcached、redis 等，数据主要存放在内存中，一般用作数据的缓存。
  - 分布式：例如 Dynamo、Riak 等，数据经划分后存放在不同的机器上，同一项数据可以在不同的机器上存有副本。为了提高系统的性能，分布式键值对存储往往牺牲了数据的一致性，采用比较弱的最终一致（eventually consistent）模型，但仍然能够满足一般应用的需求。
- 图数据库（graph database）：数据以图的形式组织，数据项是图中的顶点，每个顶点可以带属性，数据项之间的联系用边来表示，边上也可以带属性。图数据库有 Neo4j 等。
- 多模型（multi-model）：同时支持上述若干种模型，例如 OrientDB、ArangoDB 等。

一开始 NoSQL 这个词指的是这些新的数据库软件放弃了对 SQL 的支持，特别是表格之间的 join 操作，提供了更加简单的访问接口。后来也有一些 NoSQL 软件增加了对 SQL 的部分支持，因而现在 NoSQL 一般被解释为 Not only SQL。

下文将介绍 MongoDB，其他 NoSQL 软件可以查阅相关材料。

3.6.1 MongoDB 简介

MongoDB 是众多 NoSQL 数据库中比较有代表性的例子，使用简便，性能也很好，在 Web 应用领域有着广泛的使用。

MongoDB 是基于文档的存储，数据以文档为单位组织。存储的数据分为数据库（database）—集合（collection）—文档（document）三级，一个数据库可以包含若干个集合，而一个集合可以包含若干个文档。文档在表达上采用 JSON（JavaScript Object Notation）格式，而在 MongoDB 内部存储时则采用更加紧凑的二进制 BSON 格式。

文档是 JSON 格式的对象。JSON 格式的数据可以是字符串、二进制字节串、整数、浮点数、布尔值、空值，也可以是由多个 JSON 数据组成的数组，而对象则是由多个键值对构成的，其中键只能是字符串，且在同一个对象内不能重复，对应的值可以是任意的 JSON 数据。这些键值对被称为这个对象的属性，键是属性名称，而值则是属性内容。代码 6-1 是描述某人信息的一个文档。

代码 3-1 一个 MongoDB 文档的 JSON 表达

```
{
  "_id": 42,
  "name": "John Doe",
  "age": 23,
  "gender": "Male",
  "tags": ["geek", "video games", "sports"]
}
```

每个文档都有一个_id 属性，作为该文档的标识符，类似于 SQL 中的主键(primary key)。同一个集合内的所有文档的_id 属性是不同的。_id 属性可以在文档插入时由系统自动产生，也可以手工指定。集合内的文档会以_id 的值自动建立索引，同时用户也可以指定文档其他属性创建新的索引，加快以这些属性为关键字的查询。

和 SQL 相比，MongoDB 的数据模型不需要事先定义统一的数据格式，使用起来很灵活，非常适合 Web 应用快速开发的场景。比如新的需求要在人的描述信息中加入主页地址，如果用 SQL 作为数据存储，需要通过修改表的结构新增一栏，同时访问数据库的代码可能也要做相应的修改；而如果使用 MongoDB 就可以直接往文档中插入新的属性，原先的代码保持不变。此外，JSON 格式也是 Web 应用交换数据的通用格式，从 MongoDB 中取出的数据可以直接以 JSON 格式发给客户端的浏览器，而不用像 SQL 那样还需要再显式地转换成 JSON 格式。

MongoDB 的基本操作包括文档的插入、更新、删除和查询，如表 3.5 所示。值得一提的是，MongoDB 为更新文档提供了语义丰富的操作子，可以单独对文档的某个属性进行修改，还可以进行增加、倍增这样的原子操作，对于数组还可以进行插入、弹出元素的操作，而且这些也是原子的。例如， db.users.update({_id: 42}, {$push: {tags: “movie”}})就可以向前面给出的这个人增加一个 movie 标签。

表 3.5 MongoDB 的基本操作

操作	作用
insert	向集合中插入新的文档
update	更新符合条件的文档
remove	删除符合条件的文档
find	查询符合条件的文档

MongoDB 可以单机使用，也可以部署在分布式的环境中。当分布式部署时，集合中的文档会根据指定的属性值进行分块（sharding），分块后保存在不同的机器上。一个分块包

含属性值在一段范围内的所有文档，分块的范围可以重叠，这样一个文档可以保存在多台机器上，防止一台机器出现故障时导致数据的丢失。此外，系统还会动态地调整机器之间的负载，根据文档属性的分布调整分块方式，保证机器之间的负载均衡。

MongoDB 还具有一定的数据分析能力，可以对数据进行统计，也可以由客户端发送 JavaScript 的代码到服务端，执行自定义的统计功能。另外，MongoDB 还可以当文件系统来用，这项功能被称为 GridFS。

除了自己动手部署使用 MongoDB，还有像 MongoHQ 这样的 MongoDB 云服务，可以像 AWS 等服务一样直接使用，根据使用资源的量收取一定的费用。

3.7 内容回顾与推荐阅读

本章介绍了支撑大数据处理的各种系统，从传统的高性能计算，到新兴的分布式计算，以及和互联网密不可分的云计算。在分布式计算框架中，除了通用的 Hadoop 和 Spark 平台，还介绍了专门用于进行大规模图计算的框架，并说明在数据规模一定的情况下，好的单机系统甚至能获得比分布式多机系统更好的性能。此外，本章还介绍了在数据库领域兴起的 NoSQL 系列数据存储软件。

目前开源领域已经涌现出了数量众多的大数据处理软件，希望读者通过本章的学习能对这些软件有所了解，知道它们的适用问题和基本的使用方法，并能根据实际需求进行合理地选择。例如，如果要解决深度学习（deep learning）的问题，由于在处理过程中主要进行的是稠密矩阵的运算，因此传统的高性能计算会更加适合来处理。实际上，目前开源的深度学习系统几乎都是在高性能计算库的基础上进行了封装，给用户提供了深度学习领域特有的接口，而主要的计算都是由底层的计算库来完成的。

需要指出的是，系统的计算能力（或存储能力）是由硬件资源决定的，对于给定规模的问题，首先要有足够的硬件资源。系统软件的作用是对于实际的问题充分发挥这些硬件资源的能力，同时给用户提供一个友好的开发接口，提高生产率。在问题复杂性越来越高的前提下，系统软件的作用也显得越来越重要。

推荐阅读的主要著作和译著如下。

- MPI 与 OpenMP 并行程序设计（C 语言版），Michael J. Quinn（美），清华大学出版社，2004 年 1 月
- Hadoop 权威指南（第 3 版），Tom White（美），清华大学出版社，2015 年 1 月

- Spark 大数据处理技术，夏俊鸾等，电子工业出版社，2014 年 12 月
- MongoDB 权威指南（第 2 版），Kristina Chodorow（美），人民邮电出版社，2014 年 1 月

3.8 参考文献

[1] （Ghemawat, et al. 2003）Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung. " The Google file system. " ACM SIGOPS operating systems review. Vol. 37. No. 5. ACM, 2003.

[2] （Dean & Ghemawat 2008）Jeffrey Dean, and Sanjay Ghemawat. " MapReduce: simplified data processing on large clusters. " Communications of the ACM 51.1 (2008): 107-113.

[3] （Chang, et al. 2008）Fay Chang, et al. " Bigtable: A distributed storage system for structured data. " ACM Transactions on Computer Systems (TOCS) 26.2 (2008): 4.

[4] （Zaharia, et al. 2012）Matei Zaharia, et al. " Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. " Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation. USENIX Association, 2012.

[5] （Malewicz, et al. 2010）Grzegorz Malewicz, et al. " Pregel: a system for large-scale graph processing. " Proceedings of the 2010 ACM SIGMOD International Conference on Management of data. ACM, 2010.

[6] （Gonzalez, et al. 2012）Joseph E. Gonzalez, et al. " PowerGraph: Distributed Graph-Parallel Computation on Natural Graphs. " OSDI. Vol. 12. No. 1. 2012.

[7] （Kyrola, et al. 2012）Aapo Kyrola, Guy E. Blelloch, and Carlos Guestrin. " GraphChi: Large-Scale Graph Computation on Just a PC. " OSDI. Vol. 12. 2012.

[8] （Roy, et al. 2013）Amitabha Roy, Ivo Mihailovic, and Willy Zwaenepoel. " X-stream: Edge-centric graph processing using streaming partitions. " Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles. ACM, 2013.

[9] （Zhu, et al. 2015）Xiaowei Zhu, Wentao Han, and Wenguang Chen. " GridGraph: Large-Scale Graph Processing on a Single Machine Using 2-Level Hierarchical Partitioning. " Proceedings of the 2015 USENIX Annual Technical Conference. USENIX, 2015.

第 4 章

智能问答——智能助手是如何炼成的

“打开舱门，HAL!”

“很抱歉，大卫。恐怕我不能这么做。”

——《2001 太空漫游》中机器人 HAL 与大卫对话

4.1 概述

传统的搜索引擎只能根据用户输入的关键词返回匹配的网页，用户还需要进一步从这些网页中查找需要的信息。而智能问答系统则可以自动回答用户的问题，这将成为用户最贴心的智能助手。本章将着重介绍如何实现智能问答，构建智能助手。

如何变得更聪明？相信很多人都想得到这个问题的答案。然而我们也都清楚，这个问题不会有唯一的答案。但至少我们知道，判断一个人很聪明是有章可循的。最常见的办法便是问问题、做测试，根据受试者回答的正确性来评价其知识量多少、智商高低（聪明程度）。在生活中，我们经常会遇到各种问题，如果这时身边有人“上知天文，下知地理”，大家都会向他竖起大拇指。这也是各种智力竞猜类电视节目得以流行的原因之一。

在大数据时代，大量的人类知识已经被数字化。特别是随着互联网的普及、搜索引擎技术的发展，任何人只要学会使用关键词检索，便可以找到大部分自己需要的信息。从这个角度上看，“大数据”已经做到了“上有天文，下有地理”。然而在实际应用中，这种信息检索方式并不能算智能，因为这与我们通常的交流方式相去甚远。例如在王府井找一家川菜馆，我们通常需要登录餐馆信息网站，选择王府井周边这个位置范围，再选择川菜这个口味。如果用搜索引擎，则需要提取出“王府井”、“川菜”这些关键词（或者再加上“餐馆”、“餐厅”）进行检索，从结果中逐条查看网页，找到满足我们需要的结果，并从其中提取出关键信息，如餐厅名称、地址、联系电话等，如图 4.1 所示。

图 4.1 在某商业搜索引擎中检索“王府井 川菜”的部分搜索结果

与此相反，如果面对一个人（例如导游），你便可以直接问：“王府井附近有什么川菜馆？”对方直接将答案告诉你：“有家某某餐厅很不错（餐厅名称），位置就在王府井百货大楼隔壁（地址）。”如图 4.2 所示的演示应用助手，这才是最自然的交流方式。

图 4.2 在某地图助手手机应用中检索“王府井附近有什么川菜馆”界面

这个例子是智能问答技术（Question Answering）的典型应用场景。顾名思义，问答技术对于用户提出的问题予以理解，并找到答案回答给用户。这一问一答的交互方式可以极大地改善用户体验，就像本章开始机器人 HAL 与大卫对话那样，与人自然地交流。例如，苹果公司在 2011 年推出的手机应用“Siri”是一个基于问答技术的助手（如图 4.3 所示）。

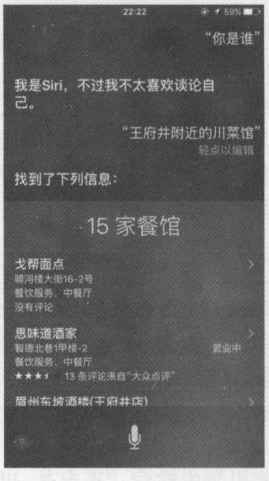


图 4.3 “Siri”语音助手界面

它可以理解多种自然语言指令，例如“给张三打电话”（拨号功能）、“提醒我明早 9 点开会”（设置日程功能）、“寻找附近的餐馆”（本地生活信息检索）等。更有趣的是，如果你问“找找附近的厕所”，它甚至还会推荐周边的麦当劳快餐馆给你。类似的手机智能助手也有类似的功能，如搜狗语音助手（图 4.4 所示）以及百度的“小度机器人”等。由此可见，应用了问答技术的智能助手让人感觉非常亲切，容易交流。值得一提的是，一些以对话为目的系统（如微软的聊天机器人“小冰”）也表现为“你有来言，我有去语”的自然交互方式，但其应答的目的不同。我们在本章主要讲述用于解答问题的“问答系统”，最后会提及用于交流的对话聊天系统。

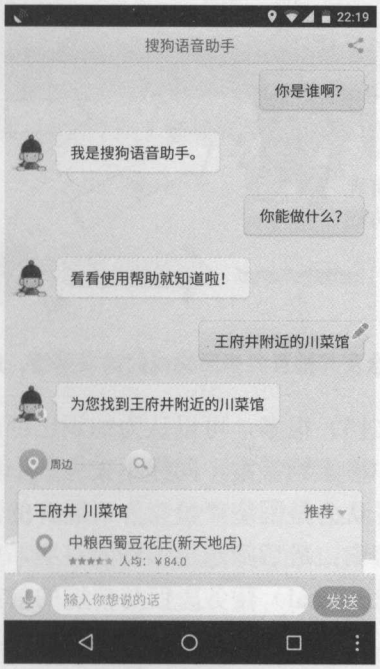


图 4.4 搜狗语音助手界面

从人类的思维逻辑上讲，对于问题的理解是基于一系列推理进行的，通过推理匹配到现有的知识，进而作出回答。例如提问：“蜜蜂有几条腿？”如果我们知道蜜蜂是一种昆虫，而昆虫有 6 条腿，那么自然可以作出回答：“蜜蜂有 6 条腿。”这种问答的思路形成了人工智能的一个重要分支：专家系统（Expert System），在 20 世纪 80 年代十分流行。在我国，亦有一些中医诊疗软件是基于这项技术编写的。显见，专家系统依赖于精确组织的知识结构（例如昆虫有 6 条腿、哺乳动物有脊椎等），又称本体（Ontology），如图 4.5 所示。我国有大量整理好的中医知识，这便是中医专家系统得以实现的原因。然而，对于那些没有组织好知识结构的门类来说，推理便无从进行。特别是人类的科学技术发展日新月异，人工

整理知识库显得越来越力不从心。因此，基于专家系统方式的问答技术已逐渐退出了主流。值得一提的是，近年来利用互联网语料自动挖掘实体关系、知识图谱的思路为这项技术注入了新鲜的血液（详见本书第 2 章知识图谱）。我们在本章后面也会看到，结构化知识仍然是问答系统的重要知识来源之一。

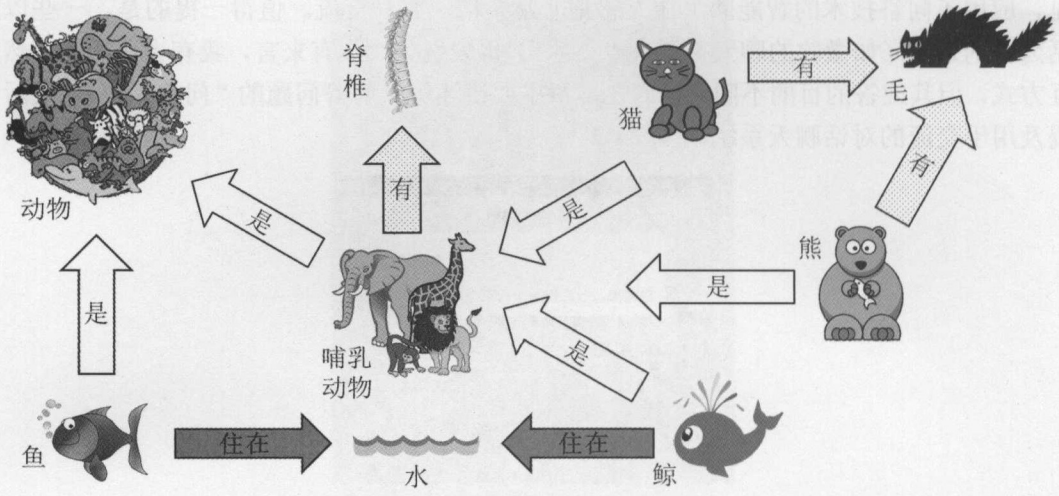


图 4.5 关于动物的概念及其相互关系所构成的语义网络，绘自（维基百科 2010）

在大数据时代，信息（文档）很多，可以认为知识已经蕴藏在这些大数据之中了，因此如果我们能从这些大数据中检索到答案，同样不失为一个好的解决方案。因此，近年来较为流行的问答系统流程可以认为是围绕“检索”而展开的，即先理解问题，知道检索什么；然后在合适的知识库中检索；最后筛选检索到的答案，整理输出。这就将问答看作是一种信息检索（Information Retrieval）任务。但与传统的信息检索（如搜索引擎）相比，用户问的不再是若干关键词，而是整句话；系统回复的也不再是若干包含关键词的文档，而是更精确的答案。可以看出，问答系统的输入部分（即问题）更不容易被计算机理解，输出部分（即答案）需要更准确。此外，答案的来源——即知识也多种多样，既有结构化的信息又有非结构化的信息，因此问答系统的难度更大。2011 年 IBM 公司推出了名为“沃森”（Watson）的人工智能系统，在美国的一个智力竞赛电视节目《危险边缘》（*Jeopardy*!）中与人类同台竞技，回答主持人提出的涵盖多种主题、学科的智力题，最终在总决赛中击败了人类选手（如图 4.6 所示）。这个系统集自然语言处理、信息检索、知识表示、自动推理等技术于一身，使用了字/词典、百科全书、新闻作品等数百万的文档，并在硬件上有足够的计算资源支撑，才取得了如此令人瞩目的成绩。与之相比，通常我们使用的问答技术虽然规模没有那么大，但其技术原理是相似的。在本章中，我们将对这类问答系统的原理进行介绍，希望能够对读者朋友有所启发。

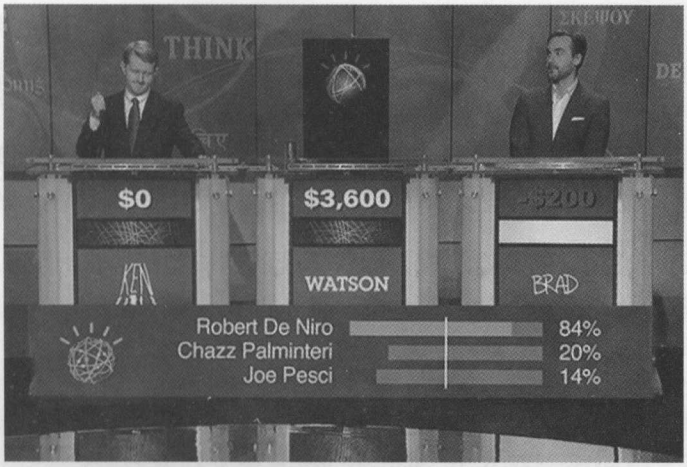


图 4.6 沃森系统在《危险边缘》竞赛节目中答题现场（Higgins 2013）

4.2 问答系统的主要组成

问答系统的基本组成，与人进行提问—思考—回答的思维过程相近，大致分为 3 个部分。

1. 问题理解：对于自然语言输入的问题，首先需要理解问题问的是什么：是在问一个词语的定义，是在查询某项智力知识，是在检索身边的生活信息，还是问某一事件的发生原因，等等。只有准确地理解问题，才有可能到正确的知识库中检索答案。例如，问题“北京的温度是多少”是在问北京这个城市的气温；而“太阳的温度是多少”则是在问一项天文（物理）知识。字面看来很相近的两句话，如果理解错误，在气象信息里寻找“太阳”这个城市的气温，则南辕北辙，无法提供答案。

2. 知识检索：自然语言提问的问题在理解后，通常会组织成为一个计算机可理解的检索式。具体检索式的格式则由知识库的结构决定。例如，如果我们采用搜索引擎作为知识来源，那么理解后的问题就可以是若干关键词；如果采用百科全书作为知识来源，那么问题就应组织为一个主词条及其属性。以“北京的面积有多大”这个问题为例，如果用搜索引擎检索，可以生成“北京”、“面积”这两个关键词；如果用百科全书，则应在“北京市”这个词条中，检索“面积”这一属性信息。

3. 答案生成：通常来说，检索到知识并不能直接作为答案返回。这是因为最精确的答案往往混杂在上下文档中，我们需要提取出其中与问题最相关的部分。例如用搜索引擎检

索到若干相关的文档，我们便需要从这些文档的大量内容中提取出核心的段落、句子甚至词语；百科全书的知识结构可能与提问并不一一对应，例如北京市的城市面积可能在不同历史时期有多个不同数值，就“北京的面积有多大”这个问题而言，我们可以取最新数值作为答案；而如果加上限定词如“建国初期”（当时北京市行政区划仅包含现城区的一部分），我们还需要针对这些约束条件，选取最佳的答案。

上面的概述是问答系统的基本流程，问答系统的结构如图 4.7 所示；但根据知识组织形式不同，问答系统还有多种不同的技术细节。下面我们就一一介绍。

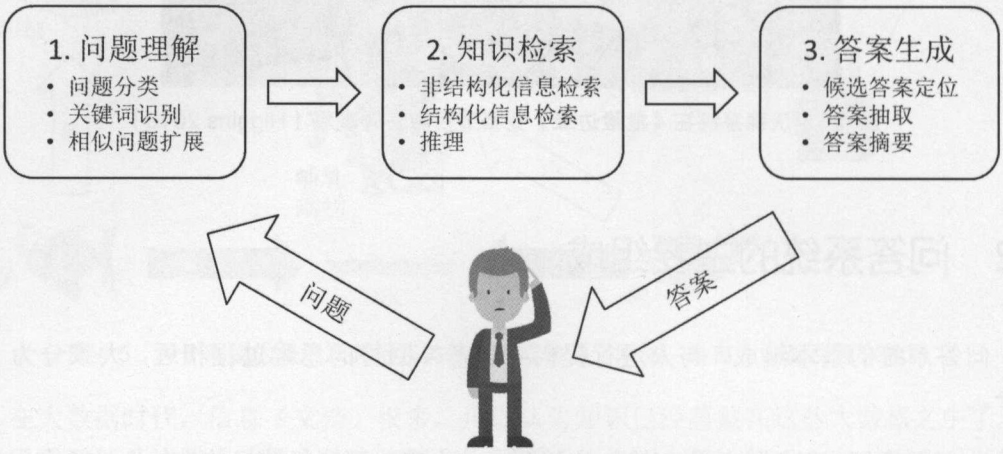


图 4.7 问答系统结构图

4.3 文本问答系统

文本问答系统是最基本的一类问答系统。其包含的模块和技术涉及问答系统的方方面面，也是各类问答的基础。下面我们就按照问答系统的 3 个基本环节来逐一展开。

4.3.1 问题理解

问题理解的核心是理解用户在“问什么”——一方面理解问的是什么事情，另一方面理解问题是什么类型。由于一个问题可能有多种不同问法，问答系统还需要进行适当的扩

展，以便找到所有相似的问法。

1. 问题理解的内容

大家都知道描述一个事件通常要包含“时间”、“地点”、“人物”等要素。对于提问来说，人的问题也无非是询问这些信息点。有的研究者把问答系统的目标定义为解答这样一个问题：

谁（Who）对谁（Whom）在何时（When）何地（Where）做了什么（What），是怎么做的（How），为什么这样做（Why）？

在英文中，问句通常由上述疑问词起始。在中文则不尽相同。然而这些基本要素的提问形式仍然是相近的。研究者们总结了提问的目标和要素，整理出了若干种分类体系（taxonomy），既有平面分类（flat），又有层次分类（hierarchical），包括如下。

- UIUC 分类体系（Li & Roth 2002）：这是一个双层的层次结构体系，主要针对事实类问题（factoid question），设计了如下 6 个大分类和 50 个小分类。
  - 缩写（Abbreviation）：缩写或缩略形式。
  - 实体（Entity）：指问题的答案是某种事物，例如动植物、颜色、货币、食物、语言、体育、科技等。
  - 描述（Description）：询问某个东西的定义、描述，某件事的原因等。
  - 人物（Human）：询问某个/某些人，人物的称号、描述等。
  - 地点（Location）：包括城市、国家、省份/州、山脉等。
  - 数值（Numeric）：包括数目、日期、距离、次序、温度、价钱等。
- Moldovan 等人的分类体系（Moldovan, et al. 1999）：这也是双层的层次结构体系，但第一层主要针对问句的形式（疑问词），第二层针对答案的类别。
  - 什么（What）：如基本的“什么”类问题，以及“什么人”、“什么时间”、“什么地点”等。
  - 谁（Who）：询问动作的施动方（主语）。
  - 谁（Whom）：询问动作的受动方（宾语）。
  - 怎么、多么（How）：根据英语的 how 词组，这个类别还包括“多少”（how many / much），多远或多长（how far / long）等。
  - 哪里（Where）：询问地点。
  - 何时（When）：询问时间。
  - 哪个（Which）：这一类会与其他类别交叉，如哪个人（who）、哪个地方（where）、哪个时间（when）等。

- 名字（Name）：这一类同样涉及其他类别，如人名（who）、地名（where）等。
- 为什么（Why）：事情的原因。

- 单层平面分类如（Radev, et al 2005）等设计了 17 个类别，包括人物、数字、描述、原因、地点、定义、缩写、长度、日期等。
- 还可以根据问题所属的垂直领域（主题）进行分类，如天气类、导航类、餐馆类等。这样做的目的是采用特定垂直领域的功能来处理相应问题，例如天气类问题则交由天气数据接口回答，导航类问题则切换至导航算法处理。

从上面的整理我们可以看出，为了理解问题，我们需要知道问题是怎么问的（疑问词）以及问题的关注点是什么。例如“泰山有多高”这个问题，问的是“泰山”这个事物的“高度”（数值）；“怎么做红烧肉”这个问题则是问“红烧肉”这个事物的制作方法（即烹饪方法）。确定了这两个关键因素，我们便可以得知用户究竟需要什么信息，以及信息的类型是什么。

2. 问题理解的方法

从自然语言提问的问题中提取出关键成分的过程主要涉及自然语言处理的语义分析技术。

最直观的做法可以采用模板匹配的策略，将同类问题的共性部分提取出来作为模板，有变化的部分自然就是查询的关键词了。例如“XXX 是什么”这个模板，可以识别一种定义类查询的句式。在用户输入问句后，如果该句能匹配上这个模板，则 XXX 的部分即为关键词。

模板匹配的优势在于逻辑清晰直观，易于理解和编写。但它的劣势也是显而易见的：形式固定，对于千变万化的自然语言不容易灵活适应——直到用户编写了相应的模板。例如，即使是菜谱查询这个简单的例子，人们在描述时就有多种问法：红烧肉怎么做，怎么做红烧肉，红烧肉的烹制方法是什么，红烧肉的制作过程……可以说，每有一种提问句式，我们就要写一条模板匹配规则。此外，在实际应用中，人们的提问可能有一些句子开头或结尾的虚词，例如“怎么做呀”、“是什么啊”以及“请问”、“我想知道”等。这些词语同样要被模板覆盖到，否则即便在人看来意思完全相同的两句话，计算机也无法“理解”。

灵活的技术则要从词法、句法的分析入手。例如将问句分词做词性标注，做句法分析，分析出主语、谓语、宾语等成分；哪些词语是名词、动词、形容词；哪些词语是命名实体（Named Entity），有重要的作用……进而移除停用词、非关键词，提取出问题的关注点及其限定词。

与模板匹配策略相比，自然语言处理技术可以更灵活地分析不同的问句，特别是基于机器学习方法在大数据（大规模语料）上训练出的语义分析模型，通常可以较准确地分析出句子及其各类变种。但一旦某些词、某些句型较为罕见，该模型仍然可能分析出错误的结果，影响后续步骤的准确性。而且这些模型并不像模板那样直观，我们不容易干涉机器

的自动处理结果。一旦出错，我们甚至不知道如何修改。此外，自然语言处理技术要求的技术储备较多，门槛高，未必适合小规模系统的快速开发和部署。

3. 问题扩展

自然语言的复杂性增加了问题理解的难度。一个问句除了可能有句式变化外，甚至还可能有同义词造成的多样性。对于不同的问题理解方法和知识组织形式，有的可能更适应句式变化，有的可能更易于理解词义。通常我们还需使用其他的自然语言分析工具来消除句子歧义，并针对相同意思扩展原始问题。例如问题“谁是贝克汉姆的老婆？”和“小贝妻子叫什么？”这两个问题没有一个词是相同的，但却表达了同样的含义。

在词的级别上，借助《同义词词林》、知网（HowNet）这样的同义词词典及词语知识图谱可以扩展我们的词库，或者从语料中学习新词的词义，如上句例子中的“贝克汉姆”别名“小贝”；在句子的级别上，借助句子复述技术（Paraphrase）可以识别同一含义的不同表达方式，如上句例子中“谁是+某人物关系”与“某人物关系+叫什么”是同一含义（汤洋 2010）。

4.3.2 知识检索

知识库直接影响了问答系统回答问题的能力和效率。一个大而全的知识库可以使问答系统更“聪明”，能够回答更多的问题，但可能降低性能，影响用户体验。因此，知识库的组织管理通常和信息检索技术密不可分。

前面提到，知识库既可以由人工整理成结构化的数据，又可以以非结构化的方式存储以便后期检索。在大数据时代，结构化的数据少而精，非结构化的数据多而全。我们可以利用这两方面的优势，从少而精的知识中提供精准答案，从多而全的数据中挖掘更有可能（例如概率更大）正确的答案，从而满足问答用户的需要。

1. 非结构化信息检索

非结构化的信息，通常是指没有或很少标注的整篇文档组成的集合。在这些文档中，信息蕴含在文本中，并没有组织成实体、属性这样的结构。这时我们可以借助信息检索技术挖掘与问题相关的信息。

最直观的理解便是使用搜索引擎。我们把问题提取出关键词，便可以查询索引，得到与这些关键词最相关的文档。再由后续的筛选和提取步骤，生成最终答案。事实上，我们可以借助商业化的搜索引擎来完成这项工作，特别是现在的很多商业搜索引擎已经具备了

一定的自然语言理解能力。像 Siri 这个产品便是采用了这样的策略：当输入的句子无法被其识别（模板未匹配中）时，它便将整句话提交给搜索引擎，并把检索到的文档集合列出来，供用户自行选择。从某种意义上讲，这种方式虽然不能直接提供准确答案，但毕竟可以减少用户输入关键词的过程，也算是一种帮助了。

使用商业搜索引擎的主要问题是商业授权许可和网络延迟。因此我们还可以自行建立索引，搭建自己的搜索引擎。现在的信息检索技术已经相对成熟，如 Lucene 等开源搜索引擎框架给开发者提供了极大便利。特别是大数据资源丰富，因此采用信息检索技术搭建索引，也是很多问答系统的必经之路。由于这里涉及的技术细节较多，读者朋友可自行参考信息检索的相关书籍。

值得一提的是，基于检索得到的文档虽然都与查询（关键词）相关，但传统信息检索任务的相关性计算方法并不一定适用于问答任务。这是因为问答任务的检索式通常已经经过筛选，因此检索出的文档应当尽量满足所有查询词的查询条件。同时，由于问答系统存在后处理步骤（即选取合适的文档和合适的答案），检索步骤得到的文档并不一定要准，而要尽可能全。

在问答系统中，如果一篇文档包含与关键词相关的答案，那么这些关键词在文档中的位置应当较为靠近，而不能分散在整篇文档中。因此常用的策略是以段落为单位来衡量，计算连续的少量段落内是否出现了所有的关键词。这样可以去除一些虽与关键词相关，但与问题答案并不相关的文档。

类似地，在挑选出的多篇文档的多个段落中，也需要找出更可能包含答案的段落或局部文本，因此也要对这些文本块进行排序。在圈定文本范围时，通常只取一个最小的窗口，使得窗口内的文本包含尽可能多的问题关键词。这个局部文本块称为“段落窗口”（paragraph window）。问答系统中的经典做法是采用标准基数排序（Standard Radix Sort）算法。排序指标通常包含以下 3 个因素。

- 相同顺序的关键词数目：按照问题中各个关键词的先后顺序，统计在段落窗口内具有相同顺序的关键词数目。
- 最远关键词间距：在这个段落窗口中相距最远的两个问题关键词，在它们之间的单词数目。
- 未命中关键词数：段落窗口未包含的问题关键词数目。

经过这一步骤，检索到的文档被提炼为若干文本块，这便于之后答案生成步骤的答案提取，使问答系统的回答更加精准。

2. 结构化知识检索

应用于问答领域的结构化知识，主要侧重于一个实体（entity）的各个属性（attribute）以及它们之间的关系。主要的结构化知识有如下类别。

- 百科类知识：传统的如百科全书，现在互联网上流行的如维基百科（Wikipedia）、互动百科、百度百科等。这些百科数据是以一个个条目（以实体为主）组成的。每个条目都有其简介、属性及其他相关信息。百科条目的属性通常清晰明了，结构性强。但其他部分均为整篇非结构化文本。例如维基百科中的“北京市”条目，结构化属性包括“面积”、“人口”、“邮政编码”等，但对其历史、交通的介绍则为非结构化文本。当然，在网络百科中，一个文本中的实体名称往往以超链接的方式标明。这对我们识别主条目引用实体的情况是有利的，便于定位答案。
- 关系类知识（本体）：前文提到了本体结构，但在实际数据表示当中，通常可以简化为关系类结构——两个事物 E_1，E_2 以及它们之间的关系 R，即三元组 (E_1, R, E_2)。这可以解决问答领域中的一些事实类问题。例如：“北京的面积是多少？”这个问题，通过问题理解，我们得知问题要找“北京”这个实体（E_1）通过“面积”这个关系（R）连接的另一个事物（E_2），那么利用关系知识（北京，面积，16,801 平方公里）则可得到答案“16,801 平方公里”。比较著名的关系类知识库有 DBPedia 和 YAGO，这些都是从维基百科中抽取并组织形成的关系结构数据库。(Di Wang 2012)

可以看出目前规模较大的关系类知识都是从百科类知识甚至非结构化知识中抽取构建的。这是大数据时代的一项知识构建工作，也吸引了许多研究者的注意。仿照这种思路，我们可以根据需求，针对特定垂直领域收集数据，自行组织成结构化知识。例如自动客服类的问答系统，我们可以从电子商务网站中获取大量的商品信息，从而解决商品类的询问和答复。例如“某某相机多少钱”，“多少价位内的羽绒服有哪些厂家生产”等问题。

4.3.3 答案生成

问答系统检索到的信息，如果结构化特性不够强，则还需要进一步地筛选过滤，提取出其中最精准的答案。这对于非结构化信息检索知识来说是必不可少的。特别是前面提到的排列出的文本块，其中很有可能包含答案。如果把整块文本返回给用户，也可以算是给出了“正确”回答，但离我们人能做出的精准回答还相去甚远。究竟哪个词、哪个短语是答案呢？

在问题理解步骤中，我们除了理解问题是在“问什么”（提取关键词）之外，还可以理解问题的类型，例如问的是人物还是数值。因此这个信息便可以用来筛选答案。因此，借

助自然语言处理技术，我们可以分析答案文本块中的词语，例如命名实体识别、词性标注等，从中筛选出更可能是答案的词语或词组。

由于问题的关键词和答案的词语之间必然存在某种联系，因此我们可以考察问题和候选答案的相似度，如问题关键词和答案词之间语义联系的远近。此外，答案与问题也可能存在句式的联系。例如问题“北京的面积是多少？”中，词语“多少”可以被替换为答案，即在答案文本中寻找类似问题句式“北京的面积是 XXX”的句子。（Allam & Haggag 2012）

随着候选答案范围的逐步缩小，我们还可以借助其他工具来验证答案的可信程度。例如采用其他的信息源（知识库），在其中检索问题（词）和答案（词）的相关性。特别是在互联网中检索答案，验证问题与答案同现的频率，也是一种简单有效的验证方法。

4.4 社区问答系统

在 Web 2.0 时代，用户产生数据逐渐增多。这些数据使互联网的信息量呈爆炸性增长，形成了“大数据”。这其中有图片分享网站，博客、微博客网站，产品评论网站等，已成为学术界、工业界以及全社会的关注热点。

在本章开始我们提到，当人遇到问题时，希望有一个无所不知的大学问家来帮他解惑。在现实中，单凭少数人是无法做到“无所不知”的。然而在大数据时代，如果能把众多网民的智慧汇集在一起，就能形成“三个臭皮匠，赛过诸葛亮”的效果。因此，社区问答（community Question Answeing，cQA）网站应运而生。国外著名的有 Quora，国内有知乎，以及百度知道、搜狗问问等网站。

社区问答网站给用户提供了公开发布问题征集解答、解答他人问题的平台。前面几节我们叙述了用计算机自动进行问答是很困难的，而在社区问答网站中，解答者是人，因此很容易理解提问题人的问题含义。利用“术业有专攻”的道理，用户可以很方便地求来解答。特别是一些“非正规”的问题如脑筋急转弯等，在百科全书中不可能找到答案，而通过网友则可以得到“解答”，虽然未必“正确”，但也可以作为一种合乎逻辑的答案（如图 4.8 所示）。

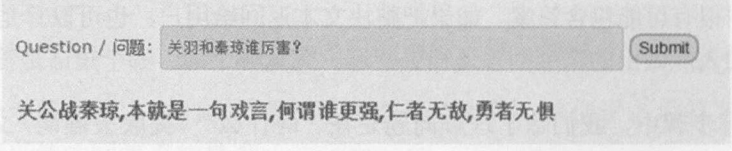


图 4.8 某社区问答系统的问题示例与回答

既然社区问答网站有很多现成的问题和答案，我们能不能用这些社区问答数据来实现计算机问答功能呢？这就是本章将要讲述的内容。

4.4.1 社区问答系统的结构

社区问答网站为我们提供了问题以及对应的答案，我们称之为“问题-答案对”，简称“问答对”（question-answer pair）。因此，与前面传统问答系统不同，此时我们已经有了问题和答案之间的联系。我们只需要找到合适的问题，再从这些问题的答案中挑出最合适的，即可完成问答任务，如图 4.9 所示。

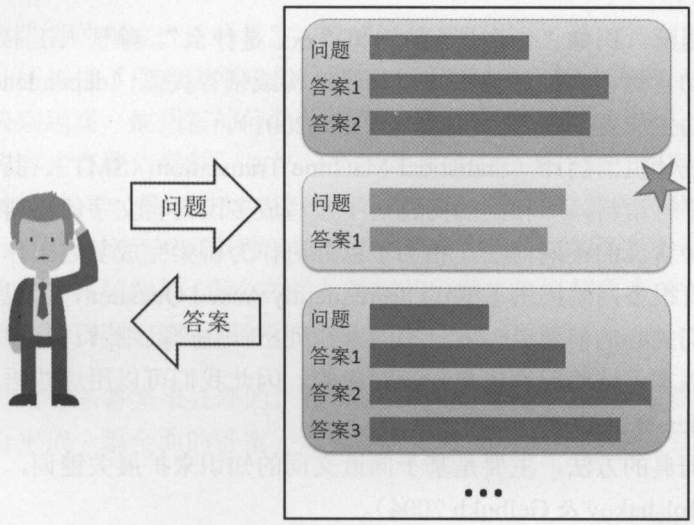


图 4.9 社区问答系统的结构示意

社区问答系统的结构可以分为以下两部分。

- 问题理解：这里的“理解”与前文含义不同，实质是在问答对数据库中，检索一个或多个与输入问题最相近的问题，作为我们“理解”了的问题。
- 答案生成：找到的相近问题对应有很多解答。但在社区问答网站中，答案的质量并不一定高。因此我们并不能直接把答案返回给用户，而要挑选出一些更有可能准确的答案，或者对多个答案进行综合，再或者对长篇答案做摘要。

可见，虽然社区问答平台为我们提供了问题与答案间的桥梁，但问题和答案自身的质量却是有噪声的。因此社区问答系统的主要难点就在于相似问题检索和答案过滤这两方面。

4.4.2 相似问题检索

*前面提到的问答系统是用问题检索知识，而在社区问答中是用问题去找问题。当问答*库较大时，我们需要对问题构建索引，这样便可以通过关键词检索到候选的相似问题。

问题的相似性问题与问题扩展所解决的问题是类似的，同样需要词义的扩展、句式的扩展。但是问题扩展是从一个原始问题生成多个候选问题；而这里的问题相似性衡量是在初步检索到候选相似问题之后进行的，因此计算规模大大减小。我们只需要在这些候选相似问题中找出最接近的一个或几个问题即可。

问题相似性度量有以下几种常见方式。

- 模板匹配，例如“什么是 xxx”和“xxx 是什么”。除了人工书写模板，我们也可以借助自然语言处理技术，对句子结构或依存关系（dependency）进行分析，从而自动生成更多的模板（Lin & Panel 2001）。
- 基于统计机器翻译（Statistical Machine Translation，SMT）。其思路是事先找到问题的平行语料，即相同的问题句子被写成多国语言文字的语料。进而可以学习出同一种含义的不同问法，相当于以外语作为桥梁完成复述工作。我们知道，互联网上有很多网站提供了 FAQ（Frequently Asked Question，常见问题解答）栏目，就人们关心的问题以一问一答的方式列出一些产品细节。其中有一些网站提供多语言版本，这些问答也是一一对应的。因此我们可以用这些语料来训练复述模型（Riezler, et al. 2007）。
- 基于词典的方法，主要是基于同近义词的知识来扩展关键词，从而识别相似问题句（Bolshakov & Gelbukh 2004）。
- 基于信息距离（information distance）的方法（Zhang, et al. 2007）。从问答这个应用场景来看，问句中的部分词语并不会给问题带来更多信息量，例如“你可不可以告诉我某某是什么”和“某某是什么”对于问题本身的信息量是接近的。我们可以借助信息论中的柯尔莫哥洛夫复杂性（Kolmogorov Complexity）来定义一系列语义度量，来衡量两个问题的语义相似性。

4.4.3 答案过滤

社区问答的另外一个特点是答案质量不高（如图 4.10 所示）。虽然很多网站提供了“最佳答案”这一项标注功能，供问题提出者标记出最满意的回答，但一方面最佳答案的标注率并不高，另一方面标为最佳答案的内容也未必真的是最正确的回答。因此我们还要综合

各方面因素提取出更可能正确的答案。

问题	低质量答案举例
如何找女朋友	你注定一辈子光棍
Matlab 里最小的正数是多少?	你的 IQ
Java2.0 有什么新的特征?	什么也没有
谁告诉我本拉登现在在哪里啊?	在我这，正在和我喝酒呢

图 4.10 低质量答案举例（汤洋 2010）

评估答案质量的工作主要集中在两方面。

- 根据答案提供者的权威性选择答案。权威性越高的用户，他的答案就可能越专业。一些问答社区提供了用户级别功能，回答的问题越多、被评为最佳回答的问题越多，他的级别越高，他回答的问题就越可能是正确答案。
- 根据答案内容本身评估质量。如果一个问题有多个答案，这些答案里可能都包含某些特定的关键词，那么这个关键词很有可能是正确答案的一部分。类似地，答案的长度、类别等信息也可以作为特征。当然，我们可以将两部分信息综合运用，例如用户有他的领域擅长，那么在相关类别的问题上可以侧重考虑该用户的答案，但在其他类别的问题上则不用特殊对待。

上述工作都是针对每条答案来处理的。在实际系统中，我们可以对多篇答案的关键要点提出摘要，并综合形成一篇全面的答案。这也是答案处理的工作内容之一。

4.5 多媒体问答系统

本章前面大量的篇幅都在介绍文本问答系统，即问题和答案都是纯文本内容。但我们知道，多媒体内容的表现力更强，直观，易于理解。尤其是对于某类问题如“如何做……”以及“……是什么样子的”，如果用文本回答，只能够逐步骤地、逐角度地描述。倘若我们能给一段视频或一幅图像，这些问题的解答一目了然。相信读者朋友也能够有体会，照着菜谱做菜和看视频学做菜，理解难度相差很多。特别是近年来互联网上的多媒体内容数量增长迅速，有图片分享网站、视频分享网站等，因此我们利用这些多媒体内容来解答问题是再合适不过了。反过来说，有时候我们询问一个难以用语言描述清楚的事物，想问它是什么。如果能够根据音像、视频等多媒体内容来直接提问，最为方便。这就是多媒体问答

（Multimedia Question Answering，MMQA）系统的主要目标（Hong, et al. 2012）。

*笔者就有一个亲身经历。笔者在新加坡时常听到一种新奇的鸟叫声，但这些鸟经常隐*蔽在树叶间，看不到样子。无奈之下，笔者便录了一段音频，上传到视频共享网站，然后在社区问答网站里发问，并附上该视频链接。很快便有热心网友提供了答案“噪鹃”（Asian koel）。这一事例生动反映出多媒体问答的必要性。

可以想象，多媒体问答系统与文本问答系统在结构上是相似的，只是多媒体问答系统所处理的问题、知识、答案不再限于文本，而包含了图像、音频、视频，等等。从技术角度上讲，除了自然语言处理，我们还需要计算机视觉、信号处理等多媒体技术，才能分析出多媒体所表达的内容。这些已超过了本章的范畴。在这里，我们仅对多媒体问答本身进行概要介绍，感兴趣的读者朋友可以参阅相关文献。

从问题出发，文本问答系统采用自然语言处理技术理解问题。而对于多媒体形式的问题，我们就要依靠相应的图像处理、模式识别等技术来识别其中的内容。以图像领域为例，有些读者可能已经用过一些商业搜索引擎公司提供的“以图搜图”功能。这个功能的输入和输出其实是同一种介质（图像），因此其中的特征信息是通用的，例如图像颜色、频谱等，这些低层次的特征可以满足任务。而如果以图搜字或以字搜图，就要加上对图像的内容理解。现有的图像处理技术已经可以识别图像中的物体，例如动物、建筑或更细致的人脸，但对于问答场景来说，用来提问的图像可能不够清晰、容易跟其他事物混淆。在这一粒度上，这仍然是很有挑战性的课题。

知识的来源同样是多媒体问答系统需要处理的问题。除了文本，我们还能提供图像、视频等生动的信息。但何时需要提供多媒体？何时提供文本就够用了？这些都需要根据问题、答案类型等特征来判断。例如，事实类问题如“泰山有多高”，我们提供一个数字就够了；但如果问“泰山的南天门是什么样子的”，就可能要提供图像。如果问的是“如何”（How-to）类问题，最好提供视频供人参考。因此在理解问题后，我们需要到不同的知识库中检索。

答案生成步骤与上面类似。如果只基于文本，我们可以方便地做综合、摘要。如果涉及多媒体内容，我们需要选出最有代表性的相关媒体。例如询问一个人物的介绍，我们在文本部分可以给出其生平，同时可以挑选一些该人物的代表图片或代表作品列在旁边供参考。这些策略往往取决于具体产品的需求。

多媒体问答系统尚属研究界的前沿课题，相关工作还并不像文本问答那样多。从需求看，定义类和“如何”类的问题是多媒体问答技术较好的切入点，但相关的语料仍需完善。现有的大量多媒体内容分散在多个网站中，质量参差不齐，特别是视频，反映其内容的信

息很少（通常只有标题和简要的介绍）。但从检索的角度看还远远不够。因此，对多媒体内容的理解也是制约多媒体问答系统发展的重要瓶颈。现有研究可以从某些特定领域（如新闻事件类多媒体内容）开始并逐步推广到开放领域的问答。

4.6 大型问答系统案例：IBM 沃森问答系统

2011 年 IBM 公司推出了沃森问答系统，它在智力竞猜节目上取得的骄人成绩，引起了很多人的重视。这一新闻就像当年“深蓝”机器人战胜国际象棋大师卡斯帕罗夫一样，既使计算机科学的研究者深受鼓舞，也提高了社会对人工智能、自然语言处理技术的兴趣，引发人们讨论。沃森系统综合了很多相关的处理技术，因此在本章我们加以概要介绍，与读者朋友共同分享其中的奥妙。

4.6.1 沃森的总体结构

与所有的问答系统结构相近，沃森也分为问题、知识和答案这三部分。但《危险边缘》竞赛模式并非普通的主持人提问—选手回答形式，而是主持人给出答案（线索），由选手进行“提问”。例如主持人说“美国之父，砍倒樱桃树”，选手则可以抢答“谁是乔治•华盛顿”。因此沃森针对这类问答模式进行了细致的处理，特别是在知识部分，有大量的假设、推理、综合步骤。图 4.11 是 IBM 公司深度问答（IBM 2011）研究组开发的深度问答体系结构。

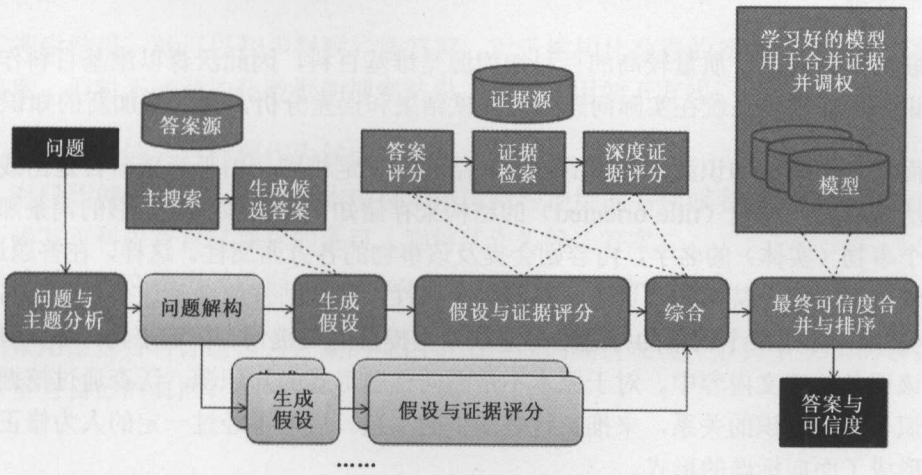


图 4.11 IBM 公司开发的深度问答体系结构，译自（IBM 2011）

4.6.2 问题解析

虽然从字面上看主持人的“提问”是“答案”，选手的“回答”是“问题”，但从语义上讲，这种问答的形式仍然不变。例如上面的例子，按照通常的思路可以转化提问为“谁是砍倒樱桃树的美国之父？”这样的问题。因此，问题的核心（焦点）仍然是需要提取的信息。此外，由于选手的回答需要显式表示这个问题的类别（“谁是……”），因此问题或答案的类型也是需要重点判断的。沃森系统使用了一套词法答案类别（Lexical Answer Type，LAT），通过提问里的一些关键词汇来推断实体类型是人、物还是地点，等等。

为了更好地理解问题，沃森用来解析语义的分析器（English Slot Grammar，ESG）专门根据竞赛节目所使用的文本进行了调整，同时还采用了谓词-论元结构（Predicate-Argument Structure，PAS）共同完成问题的解析。其中还涉及指代消解（co-reference resolution）、命名实体识别（named-entity recognition）等环节。工程师们书写了大量的规则来帮助沃森理解每一个主持人的提问。

4.6.3 知识储备

为了应对节目里各类知识的提问，沃森需要建立一个庞大的知识库。靠人工整理显然是不够的，必须要利用各种互联网资源。而且由于沃森在参加竞赛时不允许访问互联网，因此这个知识库必须事先准备好，并以适当的方式存储以便快速检索，否则无法完成“抢答”这一任务。

互联网上比较全、质量较高的百科知识源是维基百科，因此沃森以维基百科作为初始种子知识，进而根据系统在实际问题上的测试结果和误差分析，迭代增加新的知识源。

然而增加不同的知识源，知识的格式、结构不一定相同。由于竞猜节目是由线索反推事物，沃森以面向标题（title-oriented）的结构来存储知识，就像百科全书的词条那样。标题是一个事物（实体）的名字，内容则会提及该事物的各方面属性。这样，在答题过程中，如果线索可以匹配上某事物条目正文里的各个属性，那么回答就会是这个事物的名称（标题）；反过来，如果某个词条的标题恰好是线索里提及的关键词、关键事物，那么答案很可能就在该词条的正文内容中。对于原本不是面向标题形式的知识源，沃森通过挖掘内容以及该知识与其他知识的关系，来推测这些知识的主题。后期再经过一定的人为修正，全部知识都形成了面向标题的形式。

此外，沃森还挑选出被引用较多的维基百科文档内容，到搜索引擎上检索多篇网页，并将网页内容切分、重新整理并合并到原有的面向标题的文档中。通过这种方式，沃森扩充了对应条目的知识量。

值得一提的是，虽然知识都以面向标题的结构存储，但内容里的大量非结构化文本仍然不利于知识的检索。沃森的工程师们设计了一个知识抽取系统，称为“PRISMATIC”。前文提到的解析问题的语法分析器、实体识别、依存关系分析等都是由这个系统完成的。该系统建立一系列的空槽-取值关系对（slot-value pair），主要由依存关系构成。

4.6.4 检索和候选答案生成

在面向标题结构的基础上，沃森采用 3 种检索策略：一是传统问答系统的段落检索，即不限定文档内容，只检索相关词；文档搜索，按照线索涉及的属性，检索对应的整篇文档（条目）；标题搜索，按照线索提及的一些关键词检索对应的条目。同时，根据问题的分析，对不同关键词还赋予不同的权重（根据竞猜节目训练而得）。这样，通过搜索获取到相关文档，然后从文档中定位可能的答案片段。

对于传统的段落检索，沃森使用的框架基于 Indri 和 Lucene 两种搜索引擎，它们分别基于语言模型和 tf-idf；对于文档搜索，沃森使用的是 Indri 搜索引擎，搜索结果各条记录的排名和分值都将用于答案评分；对于标题搜索，则是利用维基百科建立了映射，将规范的文档标题映射到所有相同条目的百科文档上。此外，还涉及 DBpedia 这个关系本体库、IMDB 电影数据库等语义丰富的结构化知识库。

有了搜索结果，就可以初步得到候选答案。对于结构化数据的搜索结果，可以将其直接作为答案；但对于非结构化数据的搜索结果，沃森采用如下方式：

- 搜索结果的文档标题作为候选答案。
- 对段落搜索结果，提取文本中位于语义结构顶层的名词或名词短语，如果它们在维基百科中有自己独立的条目，则将其作为候选答案。
- 维基百科文档中元数据（metadata）的锚文本（anchor text）。

大量采用维基百科标题作为答案的原因是工程师们观察发现，节目中 95%的答案都在维基百科里有自己的页面。

4.6.5 可信答案确定

到这一步，候选答案已经有了初步的范围（约上千）。但要想答得准，必须要从候选答案中找出最可能正确的那一个。

沃森系统从证据出发，以其可信度来评判答案的可信度。这是通过支持证据检索（Supporting Evidence Retrieval，SER）来完成的。该方法将答案放回原始问题（线索）中，形成完整的一句话，再在搜索引擎中搜索这句话，挑选出最接近它的一些段落。这个“接近”的相关程度采用如下4种算法进行衡量。

- 段落词匹配（Passage Term Match）算法：评估问题中的关键词和段落中的关键词有多大的匹配程度。
- 二元可跳词组（Skip-Bigram）算法：尝试把问题中的关键词和段落中的关键词建立起连接。这种连接需要在语义上较为接近，即两个关键词作为语义图谱上的节点，或者相邻，或者同时连接到一个公共节点上。与上一个匹配算法不同的是，匹配算法要求词精确匹配，而本算法对于“近义词”也可以匹配，这个匹配程度的约束更宽松。
- 文本对齐（Textual Alignment）算法：直接计算包含候选答案的段落与问题的对齐程度，这就考虑到每个词的先后顺序，例如“ABC”和“BCDE”有两个词BC相对齐，但“ABC”和“CBDE”就没有两个词的对齐出现。
- 逻辑式答案（Logical Form Answer）算法：这与对齐算法相近，但并不是对齐精确匹配的词语，而引入了语法和语义的图谱，将问题和候选答案段落的图谱相对齐。当然，这种算法要求的技术难度较高，容易造成误差，因此在实际系统中，这一算法影响的比重较小。

每个候选答案都有一系列支持段落，而每个段落都被各个算法给出一个分数。只要把各个段落的同一算法打出的分数综合在一起，便可得出不同算法的评估分数，进而最后决定答案的可信程度。在沃森系统中，上述4种算法分别采用衰减和（decaying sum）、求和、衰减和与最大值作为综合的方法。

但对于评分相同的候选答案，沃森尝试图将它们合并到一起，形成完整的答案。从这些候选答案的支持段落出发，如果涉及的两个答案较为接近，则将它们合并成为一条，并选出其中较为正确的那个作为答案。这其中用到了词语形态、词语模式甚至人工构建的合并规则。此外，如果有支持度较高但答案类型错误的候选答案，沃森从原有候选答案出发，尝试寻找与之语义相近、词语间关系与提问相关且类型正确的其他答案。这样就可以找到意思相同且“答是所问”的回答。

以上就是沃森系统的基本工作流程。限于篇幅，这部分并没有介绍得十分详细。但我们可以从中看到整个系统的复杂程度。万丈高楼平地起，虽然整体复杂，但每个模块的用途明确，算法严谨，逻辑清晰。必要的时候还增加了一些人工设定的规则或模式，有些环节根据节目内容进行调优。这给了我们一些启示，即使我们搭建小规模的问答系统，也可以设计合理的结构、流程，引入必要的技术，特别是根据我们的实际需求进行详略得当的规划，便可以实现最贴心的问答助手。

4.7 内容回顾与推荐阅读

在本章中，我们介绍了问答系统的概念和应用背景，详细阐述了问答系统的主要工作原理和流程细节。对主要的几类问答系统，如文本问答系统、社区问答系统以及前沿的多媒体问答系统，本章也逐一介绍了它们的特点。我们还介绍了“沃森”这个引人注目的系统，作为问答系统的实例分析。

限于篇幅，我们只介绍了问答系统。而日渐流行的交流对话（聊天）系统与问答系统相比，更侧重于交流和应答：首先，用户的输入不一定是问题，而可能是打招呼、下指令、抒发情感等句子。从这个角度看，对话系统比问答系统更难；其次，输入的问题即使我们无法作答，也可以给出一些建议，让用户到其他地方寻找答案，或甚至老老实实承认不知道。在“图灵测试”中，机器的目标是让人分辨不出是机器还是人在作答，并非以回答正确作为检验标准。从这个角度看，对话系统比问答系统更“简单”。传统上，我们可以撰写对话模板，匹配用户输入，输出相应的回复；在大数据时代，通过挖掘网络论坛、微博回复等网民互动，可以获取更多的对话方式和对话内容，利用检索模型、机器翻译模型、深度学习模型以及情感模型，自动学习出对话过程，甚至结合情感的变化做出不同的反应。读者朋友可以参阅相关文献深入了解对话系统的原理和实现方式。

相信读者朋友能体会到，问答系统涉及的技术较多，既包含语义分析，又有信息检索，还涉及知识的挖掘与管理。的确，要想成为一个全才，势必要在方方面面都下功夫。正如同搭建系统的工作：麻雀虽小，五脏俱全。在大数据时代，信息散落在数据的汪洋之中，需要我们在每个环节都一丝不苟，认真钻研，才能挖掘出真正的宝藏。

以下是与问答系统相关的推荐阅读文献。

- Hong, Richang, et al. “Multimedia question answering.” IEEE MultiMedia 19.4 (2012): 72-78. Ferrucci, David, et al. “Building Watson: An overview of the DeepQA project.”

AI magazine 31.3 (2010): 59-79.

- Li, Xin, and Dan Roth. "Learning question classifiers." Proceedings of the 19th international conference on Computational linguistics-Volume 1. Association for Computational Linguistics, 2002.
- Tang, Yang, et al. "Information distance between what I said and what it heard." Communications of the ACM 56.7 (2013): 70-77.
- 布凡. 文本信息度量研究. [博士学位论文]. 北京：清华大学计算机科学与技术系，2013。
- 毛先领，李晓明. "问答系统研究综述." 计算机科学与探索 6.3（2012）：193-207.

4.8 参考文献

[1] （Allam & Haggag 2012）Allam, A. M. N., & Haggag, M. H. (2012). The question answering systems: A survey. International Journal of Research and Reviews in Information Sciences (IJRRIS), 2(3), 211-220.

[2] （Bolshakov & Gelbukh 2004）Bolshakov, I. A., & Gelbukh, A. (2004). Synonymous paraphrasing using wordnet and internet. In Natural Language Processing and Information Systems (pp. 312-323). Springer Berlin Heidelberg.

[3] （Ferrucci, et al. 2010）Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A. A., Lally, A., Murdock, W. J., Nyberg, E., Prager, J., Schlaefer, N., & Welty, C. (2010). Building Watson: An overview of the DeepQA project. AI magazine, 31(3), 59-79.

[4] （Gupta & Gupta 2012）Gupta, Poonam, and Vishal Gupta. A survey of text question answering techniques. International Journal of Computer Applications 53.4 (2012): 1-8.

[5] （Higgins 2013）Higgins, Chris. (2013) "What is IBM Watson?" 7 Videos from the Jeopardy! Era. http://mentalfloss.com/article/51543/what-ibm-watson-7-videos-jeopardy-era [2015-12-11].

[6] （Hong, et al. 2012）Hong, R., Wang, M., Li, G., Nie, L., Zha, Z. J., & Chua, T. S. (2012). Multimedia question answering. IEEE MultiMedia, 19(4), 72-78.

[7] （IBM 2011）IBM (2011) The DeepQA Research Team. http://researcher.watson.ibm.com/researcher/view_group_subpage.php?id=2159 [2015-12-11].

[8] （Li & Roth 2002）Li, X., & Roth, D. (2002, August). Learning question classifiers. In Proceedings of the 19th international conference on Computational linguistics-Volume 1 (pp. 1-7). Association for Computational Linguistics.

[9] （Lin & Panel 2001）Lin, D., & Pantel, P. (2001). Discovery of inference rules for question-answering. Natural Language Engineering, 7(04), 343-360.

[10] （Moldovan, et al. 1999）Moldovan, D. I., Harabagiu, S. M., Paşca, M., Mihalcea, R., Goodrum, R. A., Gîrju, C. R., & Rus, V. (1999). Lasso: A tool for surfing the answer net.

[11] （Radev, et al. 2005）Radev, D., Fan, W., Qi, H., Wu, H., & Grewal, A. (2005). Probabilistic question answering on the web. Journal of the American Society for Information Science and Technology, 56(6), 571-583.

[12] （Riezler, et al. 2007）Riezler, S., Vasserman, A., Tsochantaridis, I., Mittal, V., & Liu, Y. (2007, June). Statistical machine translation for query expansion in answer retrieval. In Annual Meeting-Association For Computational Linguistics (Vol. 45, No. 1, p. 464).

[13] （Tang, et al. 2013）Tang, Y., Wang, D., Bai, J., Zhu, X., & Li, M. (2013). Information distance between what I said and what it heard. Communications of the ACM, 56(7), 70-77.

[14] （Wang 2012）Wang, D. (2012). Learning Automatic Question Answering from Community Data. Master Thesis, University of Waterloo.

[15] （Zhang, et al. 2007）Zhang, X., Hao, Y., Zhu, X., & Li, M. (2007). Information distance from a question to an answer. In Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 874-883). ACM.

[16] （布凡 2013）布凡. 文本信息度量研究. [博士学位论文]. 北京：清华大学计算机科学与技术系，2013.

[17] （毛先领 2012）毛先领，李晓明. 问答系统研究综述. 计算机科学与探索 6.3（2012）：193-207.

[18] （汤洋 2010）汤洋. 问答社区中问题提示与答案摘要算法研究与系统实现. [硕士学位论文]. 北京：清华大学计算机科学与技术系，2010.

[19] （维基百科 2010）维基百科 (2010) 本体 (信息科学), 简单的本体示例：关于动物的概念及其相互关系所构成的语义网络. http://zh.wikipedia.org/wiki/%E6%9C%AC%E4%BD%93_(%E4%BF%A1%E6%81%AF%E7%A7%91%E5%AD%A6) [2015-12-11].

第 5 章

主题模型——机器的智能摘要利器

博观而约取，厚积而薄发。

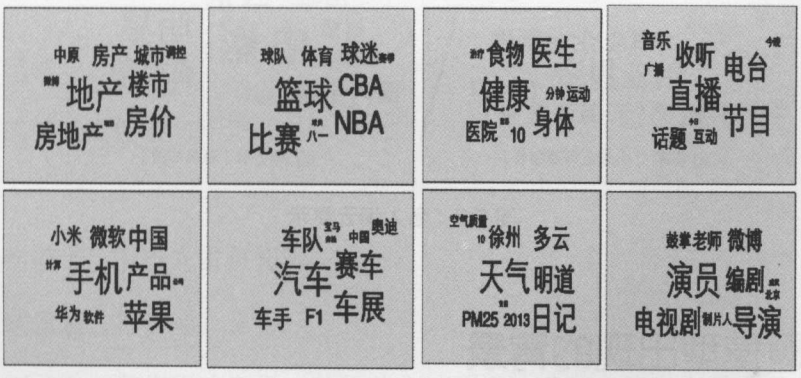
——苏轼

5.1 概述

随着互联网文本数据不断的增加，一个很重要的问题就是如何能够快速地了解和获取一个文本数据集合中主要覆盖的内容，以及如何分析每个文本文档中所包含的主要语义信息。这个问题本质上是对于文本数据集合提供了内容摘要、语义抽取和语义表示的功能。在目前这个“大数据”的信息时代，主题模型提供了一种建模思路、方法和工具，可以从大规模甚至海量文本集合中抽取主题和主题分布，其生成的结果既可以用来对语料集合进行初步的语义分析，也可以作为其他高级语义分析挖掘任务的“高阶知识”。可以这样说，主题模型在最近几年能够快速流行的一个重要原因就是在模型复杂性和解释性做了一个很好的折中，其抽取得到的主题词汇特别方便数据分析师和普通用户理解。

为了更为形象地理解主题模型，首先举一个具体的例子。假设当前任务是，要分析新浪微博上的名人账号所发表的微博内容主要涉及的语义信息，以及每个名人的兴趣分布。那么主题模型能够提供怎样的分析结果呢？

图 5.1 展现了使用主题模型获取的八个主题关键词词云。我们的语料集合是使用新浪微博所提供的前 10000 个名人在 2011 年到 2013 年两年内所发表的 2000 余万条微博。可以看到每个主题都表达了一个非常清晰的完整语义，通过主题抽取，可以很方便地获得一个语料集合上的主要语义信息，每个主题可以理解成一个在所有词汇上的权重，通过选择在一个主题内具有高权重的若干个词汇，就可以形成主题语义信息的可视化，供用户理解。



（a）经济　（b）体育　（c）健康　（d）娱乐

图 5.1　主题词云展示（4 个类别，8 个主题，每一列 2 个话题为一个类别）

从另一个方面来说，当获得了所抽取的主题语义信息之后，主题模型可以给出每个名人在各个主题上的权重分布，更具体一些，可以分析每个名人最为关注的主题语义信息。图 5.2 展示了四位名人用户的主要兴趣。为了方便展示，我们对主题打了标签，从而形成主题兴趣词云。

结合上面两个图的展示，可以看到主题模型对于大规模语料集合提供了一种有效的抽取和摘要，其输出主要包括两个方面：主题和主题分布。我们将在下面的具体内容中具体介绍主题模型。

何炅

湖南卫视知名主持人

主持人

音乐 明星 戏剧演出 明星 电视 音乐 电影 电视 电视

(a) 何炅［湖南卫视主持人］

任志强

知名地产商

互联网 股市 历史 哲理 创业投资 文学 经济 房地产 政治

(b) 任志强［知名地产商］

雷军

小米创办人，董事长兼CEO；金山软件董事长；天使投资人。

天使投资 小米手机 我们都爱用米聊

创业投资 互联网 企业管理 创业投资 手机通信 手机通信 互联网

(c) 雷军［小米公司董事长］

冯小刚

著名导演，作品有《甲方乙方》《集结号》《天下无贼》等

哲理 政治 电视 电影 明星 戏剧演出 国内新闻 电视

(d) 冯小刚［著名导演］

图 5.2 名人词云展示

5.2 主题模型出现的背景

要介绍主题模型，一个很好的思路就是先回顾一下常用的文本表示方法的发展历程。截止到目前，科研工作者和工业界广泛使用的两大类文本表示模型为：（1）矢量空间模型

（Salton, et al. 1975）；（2）统计语言模型（Ponte & Croft 1998）。虽然后续有更复杂的表示模型和方法，这两类模型仍然是最为常用的文本表示模型。这两大类模型基本的出发点，都是认为文档都是在词汇空间上进行表示的，也就是说一个文档会形成一个“文档到词汇”的映射或者表示。尽管这两种表示方法简单，但是具有很多的优点，如容易被理解、实现简单、效果稳定，等等。

如果使用线性空间来解释矢量空间模型，则可以把每个词汇对应到空间中的一个坐标轴方向，进而文档表示可以理解为确定每个坐标轴对应的坐标值。这里隐含地使用了“词与词之间是独立的”这一假设，也就是说认为词汇对应的单位向量是线性无关的；与向量空间模型类似，一元语言模型将一个文档表示成一个词汇集合上的概率分布，每一个概率值代表该文档中对应词汇的生成概率，通常这一概率值代表了文档与词汇之间的相关度。在一元语言模型中，隐含地使用了不同单词之间的独立性。

假设词典为{A_1 A_2 B_1 B_2 C_1 C_2 C_3}，第一个文档词序列为“A_1 A_2 A_1 A_2”，第二个文档词序列为“C_1 C_2 B_1 B_1”，则如果使用一元语言模型并且使用简单最大似然估计，可以将这两个文档分别表示为{A_1:0.5 A_2:0.5 B_1:0 B_2:0 C_1:0 C_2:0 C_3:0}和{A_1:0 A_2:0 B_1:0.5 B_2:0 C_1:0.25 C_2:0.25 C_3:0}。

很快，研究学者发现了这两种表示所使用假设中的问题，也是促使第一个主题模型问世的主要原因。那就是词的“同义与多义”问题。同义指的是不同的词汇在一定背景下具有相同的意思；多义指的是一个词汇在不同的背景下有不同的意思。

下面举一个具体的例子。

同义：

今天面试就是去打酱油。

今天面试就是随便参与一下。

歧义：

中午要吃饺子，下班先去打酱油。

今天面试就是去打酱油。

在这个例子中，我们不难发现，传统的文档到词的表示方法，实际上很难刻画这种词的同义与多义问题。于是人们开始思考，传统的表示模型到底哪里不好，应该如何改进呢？

5.3 第一个主题模型潜在语义分析

主题模型的精髓思想实际上都是源自于潜在语义分析（Latent Semantic Analysis, LSA）（Deerwester, et al. 1988）。潜在语义分析，打破了以往人们对于文本表示的一个限制：文本必须表示在词汇空间。

潜在语义分析创新地引入了语义维度，语义维度是文档集合上相同、相关信息的浓缩表示。如果将每一个维度对应到表示空间中的一个轴，则形成了一个语义维度空间，进一步可以将文档表示在语义维度空间。形象地说，在以往“文档→词汇”映射表示中，引入了一个语义维度，即“文档→语义→词汇”。潜在语义表示本质想法是考虑词与词在文档中的共现，然后通过线性代数的方法来提取出这些“语义”维度，然后实现文档在语义空间上的降维表示。提到了降维表示，这里值得说的是“降维”不是主题模型的最初目的，主要目的仍然是发现隐含的语义模式：因为学习得到的语义维度的数量往往都远小于词典空间中词汇的数量。因此简单地把主题模型理解为一种文档的降维表示方法并不是非常合适的。

仍然使用上面的例子。假设词典为$\{A_1\ A_2\ B_1\ B_2\ C_1\ C_2\ C_3\}$，第一个文档词序列为“$A_1\ A_2\ A_1\ A_2$”，第二个文档词序列为“$C_1\ C_2\ B_1 B_1$”。这里假设 A_1 和 A_2 表示话题一，B_1 和 B_2 表示话题二，$C_1\ C_2\ C_3$ 表示话题三。LSA 试图自动从文本语料中自动学习得到这三种隐含话题，继而将文档表示在这 3 个话题上面：文档 1 与话题一具有紧密联系，而文档 2 与话题二、话题三具有紧密联系。

下面简要介绍一下潜在语义索引，通过它我们来介绍一些主题模型的核心思想。潜在语义索引的英文全称为 Latent Semantic Indexing（LSI），这里我们大概解释一下 indexing 的意思。LSI 在提出的时候就是为了解决检索中语义不匹配问题（例如，歧义和多义），而检索算法一般是基于倒排索引进行设计的，因此沿用了检索里面的术语，叫做 Latent Semantic Indexing，LSI 又被称为 LSA，也就是（Latent Semantic Analysis）。在 1988 年，Deerwester et al.发表了 LSI 的最原始的学术论文（Deerwester, et al. 1988），尽管它的诞生和因子分析和主成分分析有着很紧密的联系，但是这并不是本文的重点，并且因子分析和主成分分析并没有直接涉及文本背景，因此这里略去不说。假设语料集合中，文档的数量为 n，词汇的数量为 m，给定一个 term-doc 矩阵 $\boldsymbol{A}(m\times n)$，其中 $\boldsymbol{A}_{i,j}$ 表示表示词 i 在文档 j 中的权重（例如设置为出现的词频数，也可以使用其他的方法，例如 tf-idf），这样矩阵 $\boldsymbol{A}$ 的每一行对应一个词汇，而每一列对应一个文档。LSI 接下来对于 $\boldsymbol{A}$ 做奇异值分解（SVD）。这里我们简要介绍一下的最后的分解结果

$$\boldsymbol{A}=\boldsymbol{TSD}^{\mathrm{T}}$$

$$T^{\mathrm{T}}T=I_r \quad D^{\mathrm{T}}D=I_r$$

$$S_{1,1}\geqslant S_{2,2}\geqslant\cdots\geqslant S_{r,r}>0 \qquad S_{i,j}=0 \text{ 当 } i\neq j$$

通过奇异值分解，矩阵 $\boldsymbol{A}$ 可以分解为三个矩阵：T，S 和 D。T 是一个 $m\times r$ 的词汇向量矩阵，S 是一个 $r\times r$ 的对角阵(对角元素递减排列)，D 是一个 $n\times r$ 的文档向量矩阵。其中 $r <= \min(m,n)$。LSI 做了降维的近似处理

$$\boldsymbol{A}\approx\boldsymbol{A}_K=\boldsymbol{T}_K\boldsymbol{S}_K\boldsymbol{D}_K^{\mathrm{T}}$$

通过这一近似处理，实际上只保留了 S 中最大的 K 个对角值（也就是奇异值），进而文档矢量矩阵 D 和词汇矢量矩阵 T 都被缩成了 K 列。其中词汇矢量矩阵的每一列就是一个主题，而文档向量矩阵的每一行就是就是一个文档对应在这 K 个主题向量上的系数表示（S 矩阵对应对角元素进行加权）。因此，给定一个原始的文档向量（也就是 $\boldsymbol{A}$ 的一列）$\overrightarrow{D_j}\approx\sum_k s_{k,k}d_{j,k}\overrightarrow{t_k}$。对于多个文档，这 K 个主题向量是共享的，但是线性结合系数是文档特定的。通过这样的表示，可以清晰地看到，每一个文档向量可以近似表示成主题向量的线性加权，也就是每一个文档都表示成了主题向量上的权重分布，也就是建立起来了“文档→语义”的关系；从另一个方面来看，一个主题矢量是在原始词汇空间上的一个向量，每个维度的数值表示该主题内对应该词汇的权重，一个词汇的权重越大，表示在该主题内部越具有代表性。尽管主题模型的数学刻画方法有很多种（矩阵分解、概率模型等），总结起来，主题模型最基本的思想可以总结为：

1．找到一系列语义“独立”的主题（在 LSI 中为线性无关的矢量）；

2．将文档表示成主题上的权重分布；

3．每个主题内部，词汇可以按照与主题的相关度进行排序，进而形成主题信息的可视化理解。

回到最开始的问题，为什么 LSI 能够解决同义和多义的问题。

同义：LSI 以及后面要提到的概率主题模型本质上就是挖掘词汇与词汇在文档层面的共现模式（可能主题模型的变种未必是文档层面上的共现）。如果两个词汇经常共现，那么他们很有可能具有相同的语义；进一步，如果两个词汇经常与一些相同的背景词汇共现，那么它们有可能具有相同的语义。主题模型通过捕捉这样的共现模式，使得最后出现在同一个主题内部、具有高权重的词汇聚合在一起。而这些聚合在一起的词汇实际上很有可能就是一些同义或者语义相近的词汇。因此，主题模型可以刻画同义现象。

多义：多义是指同一个词汇在不同背景下可能具有不同的语义。为了理解这一问题，

我们去研究 LSI 中每个主题矢量。刚刚介绍过，主题模型倾向于把相同语义的词汇聚合到同一个主题内部（注意，这是“软”聚合，而不是“硬”聚合，因此一个词汇会出现在多个主题内部。在每个主题内部，同一具有不同的权重）。当仅仅给定一个多义的词汇，很难判断这个词的具体语义，但是如果给了一个词的背景信息（如前后的词汇），则能够比较容易判断这个词对应的具体语义。将词汇聚合成主题后，给定一个词汇的时候，实际上可以去观察它所在主题内部的其他词汇，这些词汇可以帮助我们来理解这个词汇；当从一个主题换到另一个主题的时候，实际上相同的一个词汇就有不同的背景主题词汇，借助所在主题的背景词汇，就可以更为准确地判断每个词汇特定背景下的语义。

在上面的内容中，我们已经解释了同义和多义的问题，但是这只是最原始 LSI 的动机，现在的主题模型已经远远超过了这两个优点，我们会在后续给予介绍。

5.4 第一个正式的概率主题模型

LSI 在映射表示中，引入了一个语义维度，即“文档→语义→词”，然后通过线性代数的方法来挖掘词汇之间的共现关系，提取出“语义”维度。还可以使用其他方式来刻画这种思路。。随着概率统计分析在文本建模应用的不断发展，潜在语义分析从线性代数的分析模式被进一步提升到概率统计的展现模式，pLSI 或者 pLSA［4］。原始的 pLSI 论文没有用到 topic model 这一个专业术语[1]，而称主题模型为 aspect model。在很多情感分析领域内部，我们经常看到 aspects 而不是 topics，这是情感分析领域内部对于主题的特定叫法。

之前每个语义维度对应一个特征矢量，在概率模型中，每个语义维度 t 则对应到一个词典 V 上的概率分布，即{Pr(w|t)}；文档对于每个语义维度的权重，对应到概率模型中，将每个文档 d 表示成一个语义空间上 T 的概率分布，即{Pr(t|d)}。所以截止到现在，不难发现 pLSI 就是 LSI 的一种概率呈现。在原始的 pLSI 的论文中，作者清晰地讲解了 pLSI 和 LSI 之间形式化上的对应联系，在这里不予详述。我们这里只是再强调下模型假设。在 LSI 里面，我们假设主题向量之间是正交的；对应到 pLSI 里面我们做的假设是，不同主题变量之间是相互独立的。下面给出 pLSI 的一个图模型示意图，如图 5.3 所示。

1 对于这个模型有 pLSI 和 pLSA 两种叫法，其中 pLSI 中的 I 是“indexing”缩写。因为最早 LSI 是在检索背景下提出的，所以 pLSI 沿用了之前的叫法。但是随着后续工作的开展，其实 pLSI 已经不局限于检索问题，所以 pLSA 更科学一些。

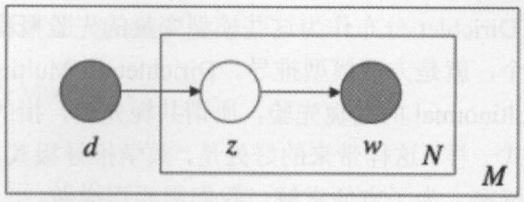


图 5.3 pLSI 示意图

在这里我们简要介绍一下上述图模型的具体含义。每个矩形方框代表重复生成过程若干次，方框右下角的字母表示重复次数；每个圆圈代表一种变量，黑圆圈表示已知变量，白圆圈表示隐含变量。箭头表示变量间的依赖关系。从图 5.3 可见，每一个词汇的生成过程，就是从 d 到 w 的一个路径“d→z→w”。其中 d→z 的生成过程是为 w 根据概率 P(z|d)选择一个主题标签；z→w 的生成过程根据已经生成的主题标签 z 来根据概率 P(w|z)来生成 w。我们会在后面的内容中详细介绍主题模型的生成过程。

5.5 第一个正式的贝叶斯主题模型

尽管 pLSI 采用了概率模型作为刻画方法，但是它并没有“将概率表示进行到底”，形式化地说，它不是一个完整的贝叶斯模型：其中的{Pr(w|t)}和{Pr(t|d)}都是直接根据数据估计出来的，都是模型参数，而没有进一步对于这些参数引入先验。

在这种背景下，2004 年，David Blei 首次提出全贝叶斯版本的 pLSI，并且将其称为主题模型，英文名叫 Latent Dirichlet Allocation（Blei, et al. 2003）。LDA 对于 pLSI 做出的改进，可以归结为以下几点：

- 实现了 pLSI 的一个全贝叶斯视角的模型和解释；
- 提出基于变分法的模型推导方法；
- 第一次显示地提出 topic model；
- 将原始 pLSI 中文档与文档、词与词之间的独立假设（bag-of-word 假设），使用了可交换性（可以简单理解为条件独立）进行解释。

但是，LDA 并没有提出主题模型思想上的改进，其生成过程基本上和 pLSI 保持了一致。其模型上所做的主要贡献，可以概括为把{Pr(w|t)}和{Pr(t|d)}这些 pLSI 中的参数看做模型变量，进而为其增加了 Dirichlet（狄利克雷）先验。大家可以注意到，Latent Dirichlet Allocation 中包含的一个主要词汇就是 Dirichlet，在此不详细地介绍该概率公式，只是从想

法上说一下为什么使用 Dirichlet 分布作为这些模型变量的先验概率函数，而不是其他的概率函数。其原因只有一个，就是方便模型推导。Dirichlet 和 Multinomial（多项式）是一对好兄弟：Dirichlet 是 Multinomial 的共轭先验，所谓共轭先验，指的就是后验的概率函数形式和先验的概率函数形式一样。这样带来的好处是，数学推导极其方便。所以采用 Dirichlet 先验并没有什么神奇的秘密，为了方便求解，我们需要那样做。

尽管来自 LDA 的创新点并不多，但是它着实带来了很多好处。容易引入更多的信息和对模型进行拓展，如引入作者、时间维度。除了这些显而易见的好处外，其带来本质的优势，就是借着贝叶斯这棵大树，大踏步地成长起来：通俗地说，所有贝叶斯的相关技术和研究成果都可以套用在 LDA 这个模型上。所以当贝叶斯火得一塌糊涂的时候，LDA 又怎么能不火？从另一个角度来说，单从模型估计和数据拟合上来说，pLSI 确实有着一些弊病，如过拟合数据、还有“零频率”问题（对于领域外词汇无法处理）。从第三点来说，LDA 是一个想法简单、却又足够灵活的模型，当图像等其他非文本领域的研究学者看到，非标准文本数据与文本数据在使用 LDA 并没有本质差别的时候，也加入了主题模型的应用大军。

5.6 LDA 的概要介绍

LDA 是一种层次的贝叶斯模型。为了方便以后的叙述，我们下面详细地介绍一些形式化定义作为基础。假设整个文档集合一共有 T 个主题，每个主题 z 被表示成一个词典 V 上的一元语言模型 θ_z，即词典上的一个多项式分布。进一步假设每个文档 d 对应这 T 个主题有一个文档特定的多项式分布 d。图 5.4 展示了 LDA 的生成过程。

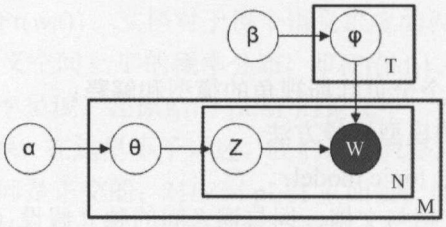


图 5.4 LDA 生成过程

一个文档的生成过程如下。

- 采样 $\theta_d \sim Dir(\alpha)$
- 对于文档 d 中的每一个词 w，我们：

——采样一个主题标签 $z \sim \text{Multi}(\theta_d)$；

——生成对应的 $w \sim \text{Multi}(\varphi_z)$。

其中 $\varphi_\cdot \sim Dir(\beta)$

对于初学者来说，可以只关注“文档→主题→词汇”这条生成链，从中提取两种概率：1）一种是“文档→主题”的分布；2）另一种是“主题→词汇”的分布。因为这两种分布都是使用多项式分布来刻画的，因此生成过程，大家可以类比抛一个多面体的色子，我们有两种不同类型的色子：第一种是“文档-主题”色子，每个文档对应这样一个色子，色子的每个面表示一个主题标签，每一个文档有不同的色子面生成概率；第二种是“主题-词汇”色子，每个主题对应这样一个色子，色子的每个面表示一个词汇，每一个主题有不同的色子面生成概率。

基于这些类比，一个词汇的生成过程就可以理解为抛两次色子的过程，第一次抛“文档→主题”色子选择一个主题标签，第二次抛“主题→词汇”色子，根据已经选择的主题标签来生成对应的词汇。

下面具体介绍一下什么是主题（topic）以及 LDA 中文本的生成过程。主题是语料集合依赖的，也就是说给定不同语料，它们背后隐藏的语义是不同的。主题是语料集合上语义的高度抽象、压缩表示。表 5.1 是几个主题的例子，我们可以看到每一个主题对应一个比较一致的语义。

在主题模型中，每个主题被表示成一个多项式分布，在实际应用中，往往截取前 10 个关键字作为结果展示。每个主题相对文档表达的内容形成了更加浓缩的表示，由表 5.1 列出了五个样例主题。

表 5.1 对文档内容进行潜在模式挖掘的主题实例

口头语	足球	电视剧	教育	健康
回复	足球	电视剧	老师	健康
呵呵	比赛	卫视	学生	医生
支持	球迷	演员	同学	身体
谢谢	体育	后援会	学习	事务
嘻嘻	球队	导演	学校	医院
偷笑	球员	杀青	大学	锻炼
快乐	女足	拍摄	教育	运动

续表

不错	联赛	拍戏	教授	治疗
感谢	曼联	剧组	培训	营养
分享	赛季	影视	课程	减肥

进一步，如果将一个名人所发表的所有微博聚合成一个文档，我们就可以得到该文档对应的主题分布，也就是这个名人的兴趣分布。假设我们这里只有表 5.2 中的五个主题，那么我们可以获得如下的名人兴趣分布。

表 5.2 名人在五个主题中的兴趣分布

名人	口头语	足球	电视剧	教育	健康
某电视演员	0.2	0.01	0.7	0.04	0.05
某足球球星	0.1	0.7	0.01	0.04	0.15
某教学机构创始人	0.2	0.02	0.03	0.7	0.05

下面假设一个足球明星发了一条微博“哈哈，终于赢了这场比赛，今晚要好好休息一下”。对于每一个词汇，LDA 首先根据该足球明星的兴趣选择一个主题标签，然后根据该标签生成该词汇。如图 5.5 所示，该球星的微博主要涉及了三个主题：口头语、足球比赛和健康。该球星更有可能选择他感兴趣的主题标签，而给定一个主题后，它更有可能生成内部高概率的词汇。

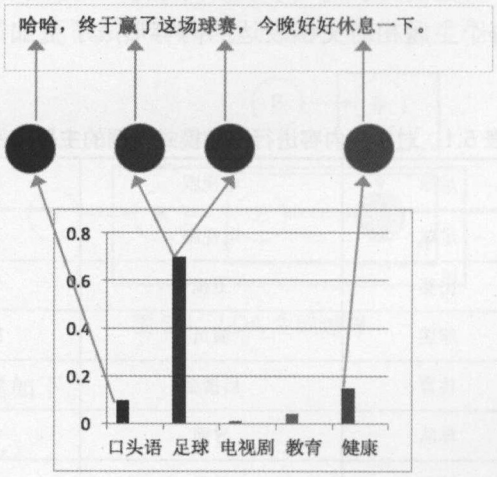


图 5.5 LDA 生成过程示例

下面着重讨论一下理解 LDA 的一些重要知识，尤其是与 pLSA 的不同点。

贝叶斯层次模型。pLSI 和 LDA 最大的区别就是 LDA 是“完全的”贝叶斯模型，而 pLSI 不是。在 pLSI 中，θ_d 和 φ_z 都是参数，因此参数的数量会随着文档的数目增加和主题数目的增加而增加，而在 LDA 中，则把 θ_z 和 φ_d 看作随机变量，都是由一组超参数来进行控制的。

可交换性。可交换性指的是，给定一个有限长度的变量序列 $z_1\ z_2 \ldots\ z_L$，对于该序列的任何一个置换 π:{1,2,..., L}→{1,2,..., L}，有 $P(z_1\ z_2 \ldots\ z_L)=P(z_{\pi(1)}\ z_{\pi(2)} \ldots\ z_{\pi(L)})$。如果一个序列中的变量满足独立等同分布(i.i.d.)，那么该序列一定是可交换的，但是反之未必成立。根据 De Finetti 定理[1]，可交换性实际指的是条件独立等同分布，即给定决定这些变量分布的一些信息后（例如，参数以及分布函数），这些变量的分布才满足独立等同分布。

概率共轭。根据贝叶斯定理 $P(\theta|X) \propto P(X|\theta)P(\theta)$，概率共轭指的是后验概率 $P(\theta|X)$和先验概率 $P(\theta)$有着同样的概率形式，当满足这一条件后，$P(\theta|X)$和 $P(\theta)$就叫做一个共轭对。在 LDA 中，狄利克雷分布和多项式分布就是一个共轭对。

先验设定。在 LDA 中，有两组先验，一种是“文档～主题”的先验，来自于一个对称的 Dir(α)；另一种是“主题～词汇”的先验，来自于一个对称的 Dir(β)。（Griffiths & Steyvers 2004）中给出了一些经验性 α 和 β 取值方法，其 α=50/T，β=0.1。由于 LDA 采用了一个完全的贝叶斯途径，对于未知文档、词汇的估计具有更强的刻画能力。实际上，pLSI 也同样可以通过采用 MAP (Maximum a Posteriori) 的估计方法来引入先验。在一般的文本挖掘任务中，这两种模型的实际效果应该接近，但是 LDA 显得更加灵活、理论基础更坚实，可以有多种模型求解方式，而 pLS 通常只能使用 EM 算法来进行参数估计，对于同样思想的模型，不同的求解方法往往会带来很大的效果上的差异。特别是当考虑的文本挖掘问题复杂，LDA 更加容易实现结合多个模型组件在一个模型中。此外还有一些非参数贝叶斯的方法引入更复杂的先验信息，例如狄利克雷过程（Dirichlet Process）（Teh 2010, Teh, et al. 2006），将会在后续章节遇到相关内容时再详细介绍。

5.6.1 LDA 的延伸理解——主题模型广义理解

上面我们已经介绍了 LDA 的基本形式，本部分从混合隐含模型的角度对于主题模型进行了更为深入的理解。

1 http://en.wikipedia.org/wiki/De_Finetti's_theorem.

混合模型的本质是基于很简单的组件模型，利用一些结合技术来进一步增强模型的表达能力，以达到通过简单模型建立复杂模型的效果；从另一方面，基于简单组件建立的混合模型，往往很容易理解，模型推导也相对清晰。通过学习共同的组件模型，可以寻找数据内部潜在的共享模式，是面对复杂问题的很好选择。LDA 本质上就是一种混合模型，包括之前的 pLSI 和 LSI 都可以理解成“广义的混合模型”。在此，我们对广义混合模型的定义是不局限于概率范畴的，而是一种建立模型和解决问题的思路与方法。

混合模型主要关注两个方面：1）组件模型的选择；2）如何结合各个组件模型。我们首先从高斯混合模型出发，介绍混合模型的基本思想。高斯混合模型假设数据的生成是由 K 个不同的高斯分布（K 个“堆”）决定的。每个数据点首先根据“堆”的先验概率挑选一个高斯分布，再由这个高斯分布的参数结构特点生成这个点。每个数据点在高斯混合模型中的概率密度函数可以写作：

$$\begin{aligned} p(x) &= \sum_{k=1}^{K} p(k)p(x \mid k) \\ &= \sum_{k=1}^{K} \pi_k N(x;\mu_k,\sigma_k^2) \end{aligned}$$

单一的高斯分布往往只能表达很简单的“圆形”（广义的圆形结构），但是高斯混合模型则可以表达很复杂的数据模式。在高斯混合模型中，组件模型是高斯分布函数，而组件的结合方式是线性的概率结合，所有数据点共享同一种线性概率系数。基于以上混合模型的理解，现在从混合模型的角度来理解 LDA。

对照高斯混合模型，在 LDA 中，我们将组件由高斯概率分布换成了多项式概率分布。多项式分布是对于离散变量概率建模的重要分布之一，是对于文本建模的首选概率分布之一。对于组件结合模式，相同的是，都是采用概率的线性结合方式；不同的就是，在高斯混合模型中，并没有文档的概念，所有的数据点都共享同一组概率系数；而在 LDA 中，则是同一个文档中的所有词共享同一组概率结合系数，也就是说概率结合系数是文档特定的。包括后续的狄利克雷和层次化的狄利克雷过程等，都是基于混合模型的思路。混合模型的重要好处是数据点可以共享组件信息和组件权重系数，这在层次化的狄利克雷过程中体现得更为明显。

同理，LSI 也可以使用混合模型的思想来理解，虽然它不是概率混合模型。在 LSI 中，组件是空间向量，组件模型结合系数则是权重系数（即文档的在主题空间的潜在表示）。

混合模型往往和隐含模型具有很深的关联。很多概率混合模型都是隐含模型，例如高斯混合模型。在隐含模型中，我们假设数据点的信息有一部分是可见的（X），另一部分

则是不可见的（Y），即隐含信息。借用期望最大化算法中的术语，我们将可见与不可见的两部分结合起来的数据（X, Y）叫做一个完整数据。例如在高斯混合模型中，一个数据点 x 和生成它的高斯分布的堆的索引 k，叫做一个完整的数据点。在高斯混合模型的原始输入数据中，我们是无法观测到一个数据点具体所在堆的索引的。在主题模型中，我们可见的变量就是词汇，而不可见变量就是每个词所对应的主题标签。将一个词汇与其对应的主题标签组合在一起，实际上就构成了一个文本中的完整数据点。对于这些隐含变量的推理和求解是主题模型求解的关键问题，一旦获得了对于隐含数据的估计，就可以进一步估计主题模型中的关键参数，如“文档～主题”分布、“主题～词汇”分布。上面所讨论的内容，正是主题模型的吉布斯（Gibbs）采样求解方法的基本思想，我们会在下一节进深入介绍。

5.6.2 模型求解

对于标准的 LDA，模型求解是一个复杂的最优化问题，很难有精确求解的方法。因此通常考虑不精确的模型求解方法。大概有两种最常用的模型求解方法：一种是基于 Gibbs 采样的方法（Griffiths & Steyvers 2004），另一种是基于变分法 EM 求解（Blei, et al. 2003）。一般来说，Gibbs 采样的方法推导起来更为简单而且求解效果也不错，所以在本文中，我们将采用其作为主要方法进行介绍。我们首先简要介绍一下 Gibbs 采样。Gibbs 采样是一个 Metropolis-Hastings 算法的特例。

其基本思想是，给定一个多维变量的分布，相比于联合分布积分，从条件分布中进行单维度采样更简单。假设我们想要从一个联合分布概率 $P(x_1, x_2, ..., x_L)$中获得 N 个样本。该方法的两个通用步骤如下。

- 随机地初始化每个变量获得 $X^{(0)}$；
  对于每个样本 $X^{(i)}$ (i=1...L)，对于每一维的变量 $X_j^{(i)}$，根据条件分布概率 $P(X_j^{(i)} \mid X_1^{(i)},...,X_{j-1}^{(i)},X_{j+1}^{(i)},...,X_L^{(i)})$，在 LDA 的基于 Gibbs 采样的模型求解中，往往采用“Collapsed Gibbs Sampling”方法（Griffiths & Steyvers 2004）。基本思想就是不考虑{θz}和{φd}两组随机变量，而只考虑对于每个词主题标签的推理求解

$$P(z_{d,i}=t \mid \vec{z}_{d,-i}, \vec{w}_{d,-i}, w_{d,i}=v) = \frac{n_{d,t}^{-i}+\alpha}{n_{d,\cdot}^{-i}+T\alpha} \frac{n_{t,v}^{-i}+\beta}{n_{t,\cdot}^{-i}+V\beta}$$

在上面的公式中，主要包括两项，第一项 $\frac{n_{d,t}^{-i}+\alpha}{n_{d,\cdot}^{-i}+T\alpha}$ 可以理解为文档 d 内部中词汇被标记为主题 t 的权重比例，而第二项 $\frac{n_{t,v}^{-i}+\beta}{n_{t,\cdot}^{-i}+V\beta}$ 则可以理解为主题 t 内部中词汇 v 的权重比例。因此，采样过程中同时考虑了文档主题概率分布和主题词汇分布概率。吉布斯采样最大的好处就是采样公式非常容易理解，而且实现方便，目前大部分有效的推理算法都是基于吉布斯采样的公式对于推理算法进行加速的。

在实际应用中，需要考虑文本数量巨大和时序演进等特征，现已有一些研究开始关注 LDA 的快速推理算法、在线学习、文本流的推理算法、分布式学习。这些研究将会使得对于 LDA 模型求解的效率大大得到提升，同时将适应文本的时序特征，可以更好地处理文本流数据。

5.6.3 模型评估

主题模型的评估长期以来都被研究学者所关注，但是从本质上并未被很好解决。主要是因为 LDA 本身是一种文本表示方法，往往很难直接评估一个表示方法的好坏。

目前有的方法，大概可以分为以下三类。

- 基于复杂度（Perplexity）的方法。复杂度经常被用在语言模型中，是用来衡量语言模型对于测试语料的建模能力的“好坏”。简单地说，复杂度是根据当前模型对于测试数据拟合程度的估计值。当一个新的主题模型被提出后，往往通过和标准的 LDA 进行测试集合上面的复杂度的对比，如果得到了更小的复杂度数值，就认为此模型的建模效果更好一些。但是基于复杂度比较的一个基本问题就是，复杂度分数小的模型是否一定在实践中获得更好的效果？更直接地说，是否能够生成更好的主题词汇？这些问题，目前还没有一个确定的研究结果。
- 基于高概率主题词的评判。每一个主题最终的表示形式是一个一元语言模型，可以根据每个主题内部词汇概率的高低来进行主题词汇的排序。得到的这些高概率的主题词汇可以直接作为输出展示给用户，然后让用户进行评估。（Chang et al. 2009）第一次系统地构建了主题模型的人工评测方法。主要考虑两个评估方面：第一个方面是主题内部一致性。具体来说，对于一个主题，首先选择具有最高概率的 5 个主题词，然后随机地添加一个在当前主题下有着较低概率但是在其他主题内部具有较高概率的词汇；第二个方面是文档内部主题分布的一致性。具体来说，对于一个文档，首先计算得到概率最高的若干个主题，然后我们随机地添加

其他一个主题。对于这两个评估方面，请人工评估找出随机添加的不相关词或者主题的难易程度。尽管这种方法简单易懂，但是需要人工判断，因此并不适用于实践。最近应用较广的是 Mimno 提出的基于主题内部高概率词汇之间的一致性的指标（Mimno et al. 2011），这种方法计算简单，通过实验证明，这种方法与人工判断的结果具有很好的关联性。

- 利用其他任务的效果来间接评估。对于一个主题模型，通过模型求解后，可以得到两种概率：第一种就是“文档～主题”的概率，第二种就是“主题～词汇”的概率。这两种概率可以直接在一些任务中使用，如文档相似度的计算、主题间相似度的计算，等等。通过这些间接任务来比较主题模型的好坏有一个问题就是，对于不同的任务，每个主题模型的优点可能不一样，所以往往一个任务不能衡量出两个主题模型之间的好坏程度。

5.6.4 模型选择：主题数目的确定

对于主题模型来说，一个非常重要的问题就是如何确定主题数目。主题数目的确定实际上是一个模型选择的子问题。与刚刚讨论过的模型评估相似，由于模型评估本身是一个非常困难的问题，所以对于主题模型中的主题数目确定仍然是一个非常困难的研究问题。目前大概有三种方法。

- 经验设定。在一些文本挖掘工作，研究人员往往通过反复调试或者枚举主题的数目来观察实验效果的好坏，例如观察高概率的主题词汇的好坏、语义是否一致，等等。这种方法虽然是启发式的，但是往往在实际中简单易行，因而是最常用的方法。实际上，大部分基于主题模型的应用工作都直接采用这种经验性的方法来设定主题数目。
- 基于复杂度（perplexity）的确定方法。在模型评估中，我们讲到了如果一个主题模型在测试预料集合上获得了较低的复杂度数值，就认为这个主题模型具有更好的模型表示能力。通过这种方法，就是对于主题数目进行枚举，然后观察复杂度的变化。但还是老问题，复杂度数值的大小和主题模型在实际任务中的好坏在理论上并不能有直接的关联。所以在一些具体的文本挖掘工作中，这种方法并不被采用，而是直接使用经验设定的方法；但是对于机器学习的研究社区，往往通过此方法来验证一个新提出的主题模型的好坏。
- 使用非参数的贝叶斯方法对主体模型进行拓展。我们将在下一节对于这个方法进行详细的介绍。其核心思想是利用一些随机过程作为先验，因此可以通过数据的自适应来自动学习出主题数量。尽管主题数量是自动学习出来的，但是需要很仔细地设

置模型因此而引入的其他超参数，所以这个方法没有本质解决主题数量的自动确定问题。非参数模型一个可能的优点就是将寻找最优的单一主题参数，变为寻找多个超参数，这样有可能使得潜在的最优参数搜索空间变大，更有可能发现最优模型。

5.7 主题模型的变形与应用

5.7.1 基于 LDA 的模型变种

随着 LDA 的推出，截止到笔者定稿时，谷歌学术搜索的显示的引用论文数高达近 2900 次。大批的学者对于基本的 LDA 在模型上进行了各种变形和拓展，还有在各种任务上的应用。在此，我们试图将这些模型的变化和拓展进行一个粗略的总结，借此来更好地了解主题模型的发展。

- 打破原有的可交换假设。在原始的 LDA 模型中，可交换性主要体现在三个方面：
  （1）文档集合内部，文档之间的顺序是没有关系的；
  （2）文档内部，词与词之间的顺序是没有关系的；
  （3）各个主题间没有关联。注意，原始的可交换假设主要是为了数学建模和求解的方便，实际上这些假设在一定程度上限制了模型的表示能力。因此，有一些模型开始对于这些可交换假设进行松弛：如引入文档间的关联（Chang & Blei 2009）、引入文档内部词与词之间的（顺序）关联（Gruber, et al. 2007）、引入主题之间的关联（Blei & Lafferty 2007）。通常情况下，一旦打破原有的可交换性后，模型的复杂度将显著增加，所以需要考虑模型表现能力与模型复杂度之间的一个权衡。另一个折中的方法，是我们并不打破这些可交换假设，而是在优化公式上加入刻画关联关系的正则化因子（Mei et al. 2008）或者利用结构化的先验信息（Chen & Liu 2014，Andrzejewski，et al. 2009）。
- 基于非参数贝叶斯方法的变形。其中比较有代表性的就是基于狄利克雷过程 Dirichlet Process,基于狄利克雷过程的方法可以自动地学习出主题的数目（Teh 2010，Teh，et al. 2006）。那么就有这样一个问题，是不是基于狄利克雷过程的方法就可以解决主题模型中的自动确定主题数目这个问题？答案是多方面的：1）在一定程度上解决了主题模型中自动确定主题数目这个问题；2）代价是必须更细心地去设定、调整一些其他参数的数值，例如超参数的设定；3）基于狄利克雷过程的方法往往在实际运行中复杂度更高，由于在学习过程中总有主题模型的产生

与消亡，这将增加模型运行和维护的复杂性。所以在实际中，往往取一个折中，看看自动确定主题数目这个问题到底对于整个应用问题的需求到底有多严格，如果经验设定就可以满足的话，那么就不用采用基于非参数的方法；但是对于一些非经验设定可满足的问题，如为了引入一些先验知识或者结构化信息，在这种情况下，往往非参数的方法是优先选择，例如树状层次的主题先验结构。还有一些其他非参数的贝叶斯方法，如基于 Pitman-Yor Process 的方法。

- 从无结构到结构化或者半结构化。标准的主题模型是一种无监督学习方法，只需要输入主题数目和一个文档集合的所有文档，模型就能够进行主题的自动学习。对于一些特定的应用问题，例如文档分类，当已经有了部分训练数据后（例如，文本的类别标签等），那么在主题模型中如何使用这些训练数据的信息。在很多情况下，可能很难直接获取训练数据，但是可以获得很多文档附加信息。纯文本往往把文本直接看做词袋子，除了词袋子，没有任何附加信息。随着文档数据格式的丰富和互联网数据的发展，传统的纯文本观点往往不适合，容易忽略一些很重要的其他特征，例如时间标签、类别标签、用户提供的标签，等等。所以在主题模型中，一个非常重要的方向就是如何在主题模型中融入这些有用的特征。在所有这些特征中，作者实体、时间、网络结构、标签信息等都是非常典型的特征，得到了学者们的广泛关注。这种信息的融入可以看做某种监督学习或者弱监督学习的方法，实际上是一种如何在主题模型中融入结构化的信息。目前主要的方法就是通过主题先验进行注入、将主题信息与响应变量进行关联、增加正则化因子，等等。

5.7.2 基于 LDA 的典型应用

随着社交媒体的不断发展，文本的形式有了很大的变化。传统的主题模型，例如 LDA，在这些新文本类型的语料上效果远没有其在传统文档集合上的效果好。因此，在此我们围绕这些新文本的特征来探讨如何应用和改进传统的主题模型。

1. 短文本

由于社交媒体是由用户所生成，很多网站为了方便用户发表观点，都支持短文本的发表，典型的应用网站如微博、电子商务网站、论坛，等等。在此，主要以微博为例来回顾一下当前短文本建模的一些研究成果。微博，由于灵活的发布消息机制以及丰富的社交关系，一经推出就得到了大量的用户使用，最近几年成为最为活跃的社交媒体平台之一。微博的短小精悍为用户发表和阅读微博带来很大方便，一般微博网站都限制短消息在 140 字节以内。对于短文本建模的主要问题就是单一文档长度过短，完全凭借单一短文本中的文本信息有的时候很难推测出整个文档的全部语义。目前最为简单而又有效的方法就是使用短文本聚合技术。

将短文本进行聚合，实际上相当于引入单一短文本的上下文背景信息，从而更好地满足主题模型对于文档内容长度的要求（Tang，et al. 2013）。（Hong & Davison 2010）经验性地将微博按照作者进行聚合成新的长文档，并且在后续实验中发现这种聚合方式是最有效的。这种聚合方式实际上是 Author-Topic 模型（Rosen-Zvi，et al. 2004）单一作者的特例，即每一个作者都唯一对应着一个文档。按照作者进行微博聚合只是一种聚合方法，还可以按照其他方式进行聚合，如首先将微博按照语义相似度进行聚类，然后每类当做一个文档；还可以根据微博中所含有的标签信息，根据标签进行微博聚合。这里的主导思想就是，我们将语义有关联的微博进行聚合，这些微博拥有一个统一的“文档”主题分布。配合着这种聚合方式，还可以对于文本片段进行切分和重组，引入各种粒度的语义信息单元，如二元组（Yan，et al. 2013）、单一短句（Zhao，et al. 2011）、段落（Titov，et al. 2008）。

第二种思路，就是引入一些外部附加的知识来丰富短文档的语义信息，如附加的标签等（Ramage，et al. 2009）。具体来说，假设这些外部的附加知识与主题分布间具有一些关联，这样，借助这些附加信息来挖掘短文本所隐含的主题信息，（Tang，et al. 2013）将这一想法进行了泛化。除了短文本自身所携带的附加知识，还可以考虑引入其他外部语义信息，如相关文档集合、知识图谱，等等。

第三种思路主要是在模型稀疏性上施加的一些限制，本质的想法就是每个文档只和一小部分主题有关，每个主题只和一小部分词汇有关（Wang，et al. 2013，Lin, et al. 2014）。这种技术主要是施加了一些模型假设，以上提及的两种稀疏性假设比较适用短文本主题建模。

2. 情感文本

社交文本另一个重要特点就是包含用户的情感或者观点词汇。随着用户在线评论数据的不断增加，消费者开始把在线评论当做一个非常重要的消费意见参考资源。因此，如何对用户评论中所表达的情感观点进行理解、抽取和摘要成为了一个非常重要的研究问题，受到了国内外学者的广泛关注（Pang & Lee 2008）。围绕着在线评论数据，科研工作者相继展开了一系列任务，从粗粒度文档层面的极性分类（Pang，et al. 2002）到细粒度的情感观点抽取（Wu，et al. 2009）。例如，一个餐馆的主题词汇可能包括食物，服务员，环境和价格，其中对应服务员的情感观点词汇可能包括友善的，粗鲁的，等等。

大概有两种方式来同时刻画主题和情感：（a）对于所有主题，设定一系列公共的情感语言模型（通常包括褒义、贬义和中性三种情感标签）（Mei，et al. 2007）。（b）对于每个主题，设定一个特定的情感模型。在联合抽取主题和观点的时候，最难的问题就是如何识别情感词汇，换句话说，如何分开主题词汇和情感词汇。第一种方法就是利用一些已经有的情感词典作为先验信息，然后在主题模型训练过程中再发现新的情感词汇；另一种方

法就是引入一些监督学习的组件到主题模型中，这样我们可同时兼有监督学习和主题模型的优点：对于监督学习来说，它比较适合区分主题和情感词汇，但是不适合用来聚类语义相近的词汇；对于主题模型来说，它比较适合聚类语义相近的词汇，但是完全无监督的主题模型不适合用来区分主题词汇和情感词汇。

除了用户评论中所包含的情感词汇，最近很多工作开始考虑引入用户的打分来改进主题情感分析。

3. 关系文本

近年来随着网络技术的不断发展，网络数据不仅仅局限于传统的文本内容，同时在“文档”间产生了丰富的链接关系，例如论文间的引用关系、用户间的朋友关注关系，等等。因此越来越多的工作开始关注如何利用链接关系来改进文本内容主题建模。实际上，链接关系本身就是一种数据类型，处于和文本同样重要的地位。我们介绍一些常见的将链接关系信息融入主题模型的方法。主要思想就是利用主题分布相似性。Relational topic model(RTM)（Chang，et al. 2009B）试图从主题分布的相似性来考虑进一步生成附加的链接结构。实际上，这是使用了一个隐含的假设:如果两个文档（或者用户）之间有着链接关系，那么他们之间的主题分布应该更为相似。在这里，他们使用分布向量的点积相似性来刻画，也就是从生成链接关系的角度考虑。除了上面刻画分布相似性的方法外，正则化技术也是一种特别常用的相似性建模方法。正则化（Regularization）是一种在最优化、统计、机器学习中经常使用的方法。其基本思想是对于模型的最优化函数上添加一些限制，通过这些限制使得模型避免过度拟合等病态学习问题。（Mei，et al. 2008，Cai，et al. 2008）提出在主题模型使用网络正则化因子。基本思想还是如果两个结点存在链接关系，那么它们之间一定存在着一些度量上的相似性。例如，如果两个作者曾经合作发表过论文，那么他们之间的研究兴趣应该相似；如果两个地方在地理位置上临近，那么这两个地方的新闻报道的话题内容应该相似。

4. 时序文本

社交文本流是用户随着时间发展而不断产生的，因此其中的时态特征非常明显，主题内容是随着时间不断变化和发展的。为了刻画这种主题的动态演变过程，动态主题模型（Dynamic Topic Model）（Blei & Lafferty 2006）假设每个时间点都对应着不同的主题 ，并且每个时间点对应的文档主题分布的先验也随着时间而变化。主要演化的假设就是基于一阶的马尔可夫假设，当前的主题信息依赖于前一个时间点的信息。后续虽然有很多延续工作，但是大部分都是以此工作进行改进的。

但是 DTM 的一个主要缺点就是不能刻画主题内容的产生、发展和消亡的动态过程。

针对这一问题，一些研究工作试图借鉴非参数模型来解决，主要是利用狄利克雷模型。在之前部分，我们已经介绍过，狄利克雷可以自动学习主题数目，可以随着语料的增多而不断生成新的主题，在新主题产生的同时，旧主题也可以随着语料的关注度减少而消亡，其内部的生成过程正好能够反映主题的动态演化过程。

另一种更为简单的技术是刻画时态主题的方法，就是除了静态的文档主题分布，对于每一个时间点设置一个特定的主题分布，然后每个文档的生成同时是由该文档的主题分布和对应的时间点主题分布所形成的。

5. 其他应用

目前主题模型广泛地应用于各种任务和领域。在此不一一详述，其他具体的应用论文可以参考（赵鑫 2011）。

5.7.3 一个基于主题模型的新浪名人话题排行榜应用

1. 系统设计背景

随着社交媒体平台的不断发展，社会媒体影响力的分析已经受到了国内外研究学者的高度关注。特别地，在社交媒体平台上，名人用户的社会媒体影响力更为重要和明显。当前很多大型社交媒体网站都提供了名人验证注册系统，这些系统认证使得名人在实际的社交活动中的身份得以证实。例如，在新浪微博上，大部分验证的名人账号来自于商业、教育、媒体和娱乐界，得到了广大微博用户的关注。通过在真实社会中的影响力和威望，通常情况下，一个名人在得到社交网站验证后，可以在很多时间快速获得大量的粉丝关注。研究名人的影响力具有重要的学术和应用价值。例如，随着互联网商业推广的发展，很多大公司也邀请名人们在社交网站平台上为其代言产品，而不仅仅局限于传统媒体。产品代言中需要考虑的重要因素就是名人的相关性和影响力。例如，给定特定领域的产品，如何对于候选的名人进行排序，并且选择合适的名人进行排序是非常重要的应用问题。

根据以上讨论，本部分主要研究关注微博平台上名人的话题排行。具体来说，给定一个内容话题后，我们对于名人进行话题相关的权威度排序。我们主要的假设是利用微博上的关注关系，如果一个名人 u 关注了另一个名人 v，则意味着用户 u 对于用户 v 的权威度认可，进一步通过引入话题信息，来学习话题相关的名人权威度。下面，我们详细地介绍一个基于主题模型的名人话题排行榜演示系统

2. 系统框架和算法设计

系统输入主要包括名人发表的微博内容和相互之间的关注关系。简而言之，包含了名人在微博平台上的文本和关系两种数据。系统输出为一系列话题内容和在每个话题内部的名人权威度排序。

整个系统处理流程主要包括四个步骤：预处理、话题抽取和名人兴趣学习、话题特定标签生成和话题权威度排序。下面分步骤详细介绍其内部实现方法，如图 5.6 所示。

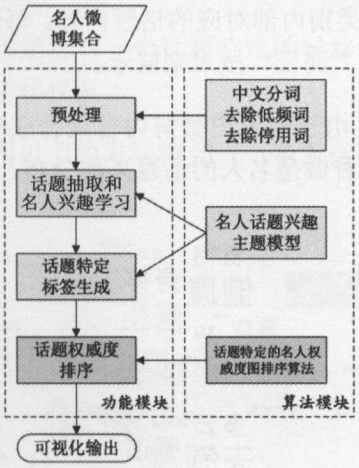


图 5.6 系统处理流程

3. 预处理

我们选择中国娱乐、科技、商业和体育四个领域的名人，通过使用名人的名字在新浪微博中进行查询，由于新浪微博已经验证了这些名人，可以很方便地找到其对应的新浪账号。接下来，通过新浪 API 爬取名人账号发表的微博、名人之间的关注关系、个人信息资料，等等。对于每一条短消息，通过 API 接口能够获得一些关键的统计量，如被转发次数、被评论次数，等等。我们限定爬取微博的时间范围为 2011 年 4 月到 2013 年 4 月。通过上述方法，我们最终收集了 13,864 位名人和对应的 26,725,958 条短消息。我们对于这些微博进行了一些基本的预处理，如分词、去停用词、低频词（<50），等等。

4. 话题抽取和名人兴趣分布学习

经过预处理，我们将每个名人用户所发表的内容打成词流。由于单一微博文档长度较短，在此，采用之前所提到的文档聚合技术，将一个名人的所有微博中包含的内容聚合成一个文档，采用“bag-of-words”来表示。为了进一步降低噪声的影响，我们对于原始的 LDA（或

者说 Author-Topic Model）进行了一个改动，引入了一个背景模型。在生成每个词汇的时候，首先引入一个二元隐变量来判断一个词汇是主题词汇还是背景词汇：如果是背景词汇，则使用背景模型来生成该词汇；如果是主题词汇，则使用主题模型来生成该词汇。

话题抽取：我们设置话题数目为 100，通过人工检查，去掉了 13 个模糊不清的话题，这样最终剩下了 87 个话题。为了展示方便，我们进一步将这 87 个话题分成了以下八个类别：微博口头语、娱乐、健康养生美容、媒体-政治、人生-感悟-运势-教育、体育、旅游、经济-科技-商业。图 5.7 展示了一个类别-话题的组织结构图。最左列是八个大类别，通过点击每个类别，就可以看到该类别内部对应的话题内容。每个话题，我们根据“话题-词汇”分布选择前十个主题词汇使用“词云”技术来展示。

名人话题兴趣分布学习：由于将用户微博内容聚合成单一文档，很容易得到“文档-话题”分布，这个分布就可以看做是名人的话题兴趣分布。

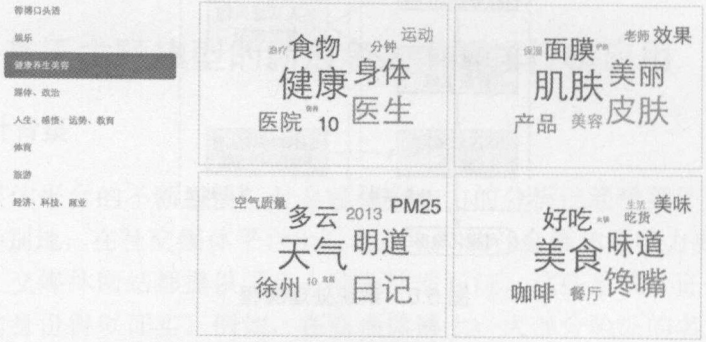


图 5.7 类别-话题的组织结构图

5. 话题特定的名人排序算法

当前系统不是对于一个名人给出一个单一的排序，而是每个话题生成一个权威度的排序。我们主要的想法就是对于传统的 PageRank 算法进行修改，主要是在算法中融入各种主题相关的信息。系统采用了（Weng et al. 2010）中所提出的 Topical PageRank 对名人进行排序。

给定一个话题 z，假设每个名人 u 具有一个话题特定的权威度 $p_u^{(z)}$

$$p_u^{(z)} \propto (1-\lambda)\sum_{v \to u} sim^{(z)}(v \to u) p_v^{(z)} + \lambda \tilde{p}_u^{(z)},$$

其中有两处引入了主题信息。首先就是先验信息的设定

$$\tilde{p}_u^{(z)} = \frac{p(z|u)}{\sum_{u'} p(z|u')}$$

其中分子就是学习得到的每个用户的话题兴趣分布，分母主要是使得先验信息可以对于名人进行加和归一化。

其次是转移概率权重的设定

$$sim^{(z)}(v \to u) \propto 1 - |p_v^{(z)} - p_u^{(z)}|$$

其中每两个用户对于每个话题都有一个话题特定的转移权重，这里引入两个用户在一个话题上的兴趣相似性来控制转移权重。

给定每一个话题，可以使用以上的方法得到一个名人话题权威度的排序，如图 5.8 所示。

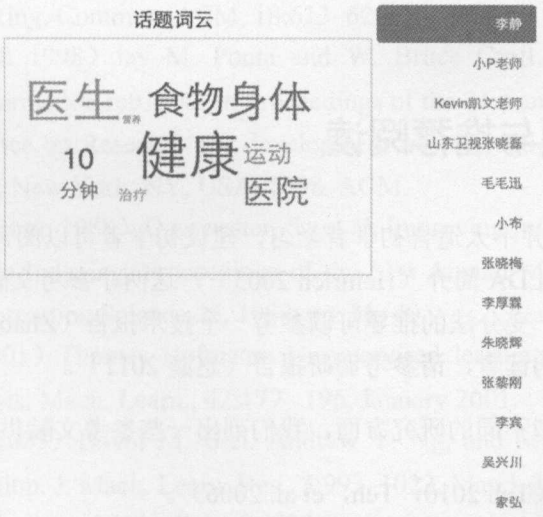


图 5.8 名人话题权威度排序

6. 名人话题特定标签生成

除了话题内容抽取和名人话题排序，系统的第三个功能是名人话题特定的标签生成。话题标签的主要作用就是可以清晰地展示出名人的兴趣。之前，我们采用 Gibbs Sampling 方法来进行主题模型的推理，一个主要的优点就是可以保存下来对于主题标签的样本。

具体来说，给定一个名人发表的内容（bag-of-words）：$w_1 w_2 w_3 ... w_N$，使用 Gibbs Sampling

方法，可以得到对应的主题标签 $z_1z_2z_3...z_N$。给定一个名人，通过这个主题标签序列，可以进一步统计一个话题内部每个词的使用频率，这样就可以对于所有名人在一个特定的话题内部所使用的词汇按照频率进行排序，如图 5.9 所示。对于每个话题，我们使用排序较高的 K 个词汇作为该名人的话题标签。

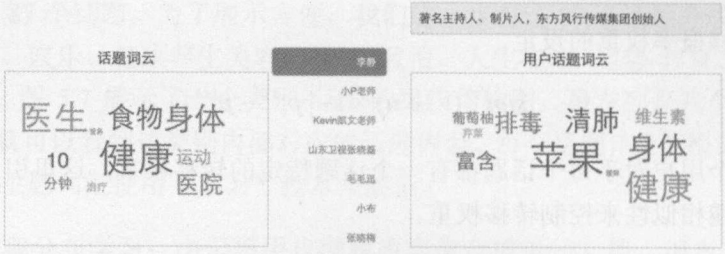


图 5.9 名人在一个特定话题内所使用词汇的排序

话题标签可以很好地浓缩展示了一个名人在一个特定话题内部发表的内容，可以方便用户了解名人的兴趣。

5.8 内容回顾与推荐阅读

LDA 的原始论文并不太适合初学者学习，建议初学者可以阅读中文的 LDA 简介（靳志辉 2013）和英文的 LDA 简介（Heinrich 2005），这两个参考文献主要讲的是 LDA 的吉布斯采样的推导，对于变分法的推导可以参考一个技术报告（Zhao 2013）。如果对于主题模型具体应用感兴趣的读者，请参考调研报告（赵鑫 2011）。

下面针对主题模型不同的研究方面，我们列出一些参考文献供读者深入学习。

- 非参数学习（Teh 2010，Teh，et al. 2006）。
- 矩阵分解（Deerwester，et al. 1988，Wang，et al. 2013）。
- （半）监督学习（Blei & McAuliffe 200）、辅助信息学习 [41]、层次化先验（Bakalov，et al. 2012）、领域知识（Chen &Liu 2014，Andrzejewski，et al. 2009）。
- 主题模型评测（Chang，et al. 2009A，Mimno，et al. 2011）。
- LDA 的快速推理和实现（Yuan，et al. 2014，Liu，et al. 2009，Yao，et al. 2009）。
- 信息检索（Wei & Croft 2006）。
- 情感分析（Mei，et al. 2007，Zhao，et al. 2010，Titov & McDonald 2008A，Titov & McDonald 2008B）。

- 短文本分析（Hong & Davison 2010，Tang，et al. 2013B，Zhao，et al. 2011B，Zhao，et al. 2011C）。
- 链接分析（Chang，et al. 2009B，Mei，et al. 2008，Cai，et al. 2008）。
- 时序分析（Blei & Lafferty 2006，Wang & McCallum 2006，Wang, et al. 2008）。
- 实体分析（Newman, et al. 2006，Xu, et al. 2009）。
- 自然语言处理：词义消歧（Boyd-Graber, et al. 2007）、句（语）法分析（Boyd-Graber & Blei 2008，Griffiths, et al. 2004）、跨语言分析（Mimno, et al. 2009，Ni, et al. 2009）、摘要（Tang, et al. 2009，Daumé III & Marcu 2006）。

5.9 参考文献

[1] （Salton, et al. 1975）G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. Commun. ACM, 18:613–620, November 1975.

[2] （Ponte & Croft 1998）Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '98, pages 275–281, New York, NY, USA, 1998. ACM.

[3] （Deerwester, et al. 1988）Deerwester, S., et al, Improving Information Retrieval with Latent Semantic Indexing, Proceedings of the 51st Annual Meeting of the American Society for Information Science 25, 1988, pp. 36–40.

[4] （Hofmann 2001）Thomas Hofmann. Unsupervised learning by probabilistic latent semantic analysis. Mach. Learn., 42:177– 196, January 2001.

[5] （Blei, et al. 2003）David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. J. Mach. Learn. Res., 3:993–1022, March 2003.

[6] （Griffiths & Steyvers 2004）T. L. Griffiths and M. Steyvers. Finding scientific topics. Proceedings of the National Academy of Sciences, 101(Suppl. 1):5228–5235, April 2004.

[7] （Teh 2010）Y. W. Teh. Dirichlet processes. In Encyclopedia of Machine Learning. Springer, 2010.

[8] （Teh, et al. 2006）Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet processes. Journal of the American Statistical Association, 101(476):1566–1581, 2006.

[9] （Chang, et al. 2009A）Jonathan Chang, Jordan Boyd-Graber, Chong Wang, Sean Gerrish, and David M. Blei. Reading tea leaves: How humans interpret topic models. In NIPS, 2009.

[10] （Mimno, et al. 2011）David Mimno, Hanna Wallach, Edmund Talley, Miriam Leenders, Andrew McCallum. Optimizing Semantic Coherence in Topic Models. EMNLP, 2011, Edinburgh, Scotland.

[11] （Tang, et al. 2013A）Jian Tang, Zhaoshi Mao, Xuanlong Nguyen, Qiaozhu Mei, Ming Zhang. Understanding the Limiting Factors of Topic Modeling via Posterior Contraction Analysis. ICML'13, 2013

[12] （Hong & Davison 2010）Liangjie Hong, Brian D. Davison. Empirical study of topic modeling in Twitter. SOMA. 2010

[13] （Rosen-Zvi, et al. 2004）Michal Rosen-Zvi, Thomas Griffiths, Mark Steyvers, Padhraic Smyth. The Author-Topic Model for Authors and Documents. Proceedings of the 20th conference on Uncertainty in artificial intelligence. UAI '04, 2004, 487–494

[14] （Yan, et al. 2013）Xiaohui Yan, Jiafeng Guo, Yanyan Lan, Xueqi Cheng. A Biterm Topic Model for Short Texts. Proceedings of the 22Nd International Conference on World Wide Web. WWW '13, 2013, 1445–1456

[15] （Zhao, et al. 2011A）Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, Xiaoming Li. Comparing Twitter and Traditional Media Using Topic Models. ECIR. 2011, 338–349

[16] （Titov, et al. 2008）Ivan Titov, Ryan McDonald. Modeling Online Reviews with Multi-grain Topic Models. Proceeding of the 17th International Conference on World Wide Web. 2008, 111–120

[17] （Ramage, et al. 2009）Daniel Ramage, David Hall, Ramesh Nallapati, Christopher D. Manning. Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora. Proceedings of the 2009 Conference on Empirical Methods in Nat- ural Language Processing: Volume 1 - Volume 1. EMNLP '09, 2009, 248–256

[18] （Tang, et al. 2013B）Jian Tang, Ming Zhang, Qiaozhu Mei. One theme in all views: modeling consensus topics in multiple contexts. KDD 2013

[19] （Wang, et al. 2013）Quan Wang, Jun Xu, Hang Li, and Nick Craswell. Regularized Latent Semantic Indexing: A New Approach to Large Scale Topic Modeling. ACM Transaction on Information System (TOIS), Volume 31, Issue 1, 2013.

[20] （Lin, et al. 2014）Tianyi Lin, Wentao Tian, Qiaozhu Mei, and Hong Cheng, " The dual-sparse topic model: mining focused topics and focused terms in short text, " in

Proceedings of the 23rd international conference on World wide web (WWW'14), pp. 539-550, 2014.

[21] （Pang & Lee 2008）Bo Pang, Lillian Lee. Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval. 2008, 2(1-2)

[22] （Pang, et al. 2002）Bo Pang, Lillian Lee, Shivakumar Vaithyanathan. Thumbs up? Sentiment Classification using Machine Learning Techniques. Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing. 2002

[23] （Wu, et al. 2009）Yuanbin Wu, Qi Zhang, Xuangjing Huang, Lide Wu. Phrase Dependency Parsing for Opinion Mining. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. 2009

[24] （Mei, et al. 2007）Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, ChengXiang Zhai. Topic sentiment mixture: modeling facets and opinions in weblogs. Proceedings of the 16th international conference on World Wide Web. WWW '07, 2007, 171–180

[25] （Chang, et al. 2009A）Jonathan Chang, David Blei. Relational Topic Models for Document Net- works. AIStats. 2009

[26] （Mei, et al. 2008）Qiaozhu Mei, Deng Cai, Duo Zhang, ChengXiang Zhai. Topic modeling with network regularization. Proceeding of the 17th international conference on World Wide Web. WWW '08, 2008, 101–110

[27] （Cai, et al. 2008）Deng Cai, Qiaozhu Mei, Jiawei Han, Chengxiang Zhai. Modeling hidden topics on document manifold. Proceeding of the 17th ACM conference on Information and knowledge management. CIKM '08, 2008, 911–920

[28] （Blei & Lafferty 2006）D. Blei and J. Lafferty. Dynamic topic models. In Proceedings of the 23rd International Conference on Machine Learning, 2006.

[29] （赵鑫 2011）赵鑫，李晓明。主题模型在文本挖掘中的应用。2011

[30] （Weng, et al. 2010）Jianshu Weng, Ee-Peng Lim, Jing Jiang and Qi He. TwitterRank: Finding topic-sensitive influential Twitterers. In Proceedings of the Third ACM International Conference on Web Search and Data Mining, 2010.

[31] （靳志辉）靳志辉. LDA 数学八卦.

[32] （Heinrich 2005）Heinrich, Gregor. Parameter estimation for text analysis. Technical report, 2005.

[33] （Zhao 2013）Wayne Xin Zhao. Varitional Methods for Latent Dirichlet Allocation. Technical report, 2013.

[34] （Yuan, et al. 2014）Jinhui Yuan, Fei Gao, Qirong Ho, Wei Dai, Jinliang Wei, Xun Zheng, Eric P. Xing, Tie-Yan Liu, Wei-Ying Ma. LightLDA: Big Topic Models on

Modest Compute Clusters. Dec 05 2014. arXiv:1412.1576v1

[35] （Liu, et al. 2009）Liu, Zhiyuan, et al. " Plda+: Parallel latent dirichlet allocation with data placement and pipeline processing. " ACM Transactions on Intelligent Systems and Technology (TIST) 2.3 (2011): 26.

[36] （Chang & Blei 2009）Jonathan Chang and David Blei. Relational topic models for document networks. In AIStats, 2009.

[37] （Gruber, et al. 2007）Amit Gruber, Yair Weiss, and Michal Rosen-Zvi. Hidden Topic Markov Models. AISTATS, volume 2 of JMLR Proceedings, page 163-170. JMLR.org, (2007)

[38] （Blei & Lafferty 2007）David M. Blei and John D. Lafferty. A correlated topic model of science. AAS, 1(1):17–35, 2007.

[39] （Chen & Liu 2014）Zhiyuan Chen and Bing Liu. Mining Topics in Documents: Standing on the Shoulders of Big Data. In Proceedings of the 20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2014).

[40] （Blei & McAuliffe 2007）David M. Blei and Jon McAuliffe. Supervised topic models. In NIPS, 2007.

[41] （Griffiths, et al. 2004）Thomas L. Griffiths, Mark Steyvers, David M. Blei, and Joshua B. Tenenbaum. Integrating topics and syntax. In NIPS, pages 537–544. 2004.

[42] （Mimno & McCallum 2008）David Mimno and Andrew McCallum. Topic Models Conditioned on Arbitrary Features with Dirichlet-multinomial Regression. UAI, 2008

[43] （Bakalov, et al. 2012）Anton Bakalov, Andrew McCallum, Hanna M. Wallach, David M. Mimno: Topic models for taxonomies. JCDL 2012: 237-240

[44] （Andrzejewski, et al. 2009）David Andrzejewski, Xiaojin Zhu, and Mark Craven. Incorporating domain knowledge into topic modeling via Dirichlet Forest priors. ICML, volume 382 of ACM International Conference Proceeding Series, page 4. ACM, (2009)

[45] （Zhao, et al. 2010）Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. Jointly modeling aspects and opinions with a MaxEnt-LDA hybrid. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 56–65, Cambridge, MA, October 2010.

[46] （Wei & Croft 2006）Xing Wei and W. Bruce Croft. Lda-based document models for ad-hoc retrieval. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '06, pages 178–185, New York, NY, USA, 2006. ACM.

[47] （Titov & McDonald 2008A）Ivan Titov and Ryan McDonald. A joint model of text

and aspect ratings for sentiment summarization. In Proceedings of ACL-08: HLT, pages 308–316, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[48] （Titov & McDonald 2008B）Ivan Titov and Ryan McDonald. Modeling online reviews with multi-grain topic models. In Proceeding of the 17th international conference on World Wide Web, WWW '08, pages 111–120, New York, NY, USA, 2008. ACM.

[49] （Zhao, et al. 2011B）Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. Comparing twitter and traditional media using topic models. In ECIR, pages 338–349, 2011.

[50] （Zhao, et al. 2011C）Xin Zhao, Jing Jiang, Jing He, Yang Song, Palakorn Achanauparp, Ee-Peng Lim, and Xiaoming Li. Topical keyphrase extraction from twitter. In ACL-HLT, 2011.

[51] （Wang & McCallum 2006）Xuerui Wang and Andrew McCallum. Topics over time: a non-markov continuous-time model of topical trends. In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '06, pages 424–433, New York, NY, USA, 2006. ACM.

[52] （Wang, et al. 2008）Chong Wang, David M. Blei, and David Heckerman. Continuous time dynamic topic models. In UAI, 2008.

[53] （Newman, et al. 2006）David Newman, Chaitanya Chemudugunta, and Padhraic Smyth. Statistical entity-topic models. In Pro- ceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '06, pages 680–686, New York, NY, USA, 2006. ACM.

[54] （Xu, et al. 2009）Gu Xu, Shuang-Hong Yang, and Hang Li. Named entity mining from click-through data using weakly supervised latent dirichlet allocation. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '09, pages 1365–1374, New York, NY, USA, 2009. ACM.

[55] （Yao, et al. 2009）Limin Yao, David Mimno, and Andrew McCallum. Efficient methods for topic model inference on stream- ing document collections. In Proceedings of the 15th ACM SIGKDD international conference on Knowl- edge discovery and data mining, KDD '09, pages 937–946, New York, NY, USA, 2009. ACM.

[56] （Boyd-Graber, et al. 2007）Jordan Boyd-Graber, David M. Blei, and Xiaojin Zhu. A topic model for word sense disambiguation. In EMNLP, 2007.

[57] （Boyd-Graber & Blei 2008）Jordan Boyd-Graber and David M. Blei. Syntactic topic models. In NIPS, 2008.

[58] （Griffiths, et al. 2004）Thomas L. Griffiths, Mark Steyvers, David M. Blei, and Joshua B. Tenenbaum. Integrating topics and syntax. In NIPS, pages 537–544. 2004.

[59] （Mimno, et al. 2009）David Mimno, Hanna Wallach, Jason Naradowsky, David A. Smith, and Andrew McCallum. Polylingual topic models. In EMNLP, 2009.

[60] （Ni, et al. 2009）Xiaochuan Ni, Jian-Tao Sun, Jian Hu, and Zheng Chen. Mining multilingual topics from wikipedia. In WWW, 2009.

[61] （Tang, et al. 2009）Jie Tang, Limin Yao, and Dewei Chen. Multi-topic based query-oriented summarization. In SDM, pages 1147–1158, 2009.

[62]（Daumé III & Marcu 2006）Daumé III H, Marcu D. Bayesian query-focused summarization. Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2006: 305-312.

第 6 章

个性化推荐系统——如何了解电脑背后的 TA

参差多态乃是幸福之本源。

——[英]罗素

6.1 概述

目前，我们已经从互联网 Web 1.0 时代跨入了 Web 2.0 的时代，以用户产生内容（User Generated Content）为重要特征的 Web 2.0 网络积累了大量的用户数据信息，包括用户在搜索引擎中的搜索历史记录、在购物网站中的购买记录和评论、在社交网站中的图片文本，等等。通过这些信息，我们可以从兴趣、喜好、消费者特质等诸多方面全面地了解网络背后一个个真实的用户，从而“投其所好”地为不同用户定制符合其需求的个性化服务，而提供这些个性化服务的一个重要渠道，就是个性化推荐引擎。在即将到来的 Web 3.0 时代，互联网和物联网将以智能化服务为核心特征，而个性化推荐技术、及其所依赖的用户理解、建模等核心构件，将成为 Web 3.0 智能网络时代的重要组成部分。

推荐系统（Recommender System，RS）的发展已经经历了近 20 年的时间，广义上的推荐系统可以理解为主动向用户（User）推荐物品（Item）的系统，所推荐的物品可以是音乐、书籍、餐厅、活动、股票、数码产品、新闻条目等，这依赖于具体的应用领域，推荐系统所推荐的物品或者对用户有帮助，或者用户可能感兴趣。

近年来，随着电子商务（E-commerce）规模的不断扩大，商品数量和种类不断增长，用户对于检索和推荐提出了更高的要求。由于不同用户在兴趣爱好、关注领域、个人经历等方面的不同，使得以满足不同用户的不同推荐需求为目的、且不同人可以获得不同推荐的个性化推荐系统（Personalized Recommender System，PRS）应运而生。目前所说的推荐系统一般是指个性化推荐系统。图 6.1 给出了一个形象的比喻：当系统检测到用户喝水的动作时，即尝试向用户个性化地推荐和准备各种可能感兴趣的后续动作。

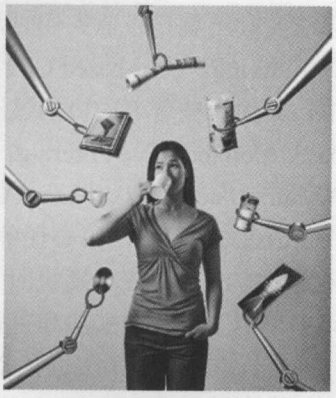


图 6.1 个性化推荐系统（注：图片来源于网络）

6.1.1 推荐系统的发展历史

如果追根溯源，推荐系统的初端可以追溯到函数逼近理论、信息检索、预测理论等诸多学科中的一些延伸研究。

推荐系统成为一个相对独立的研究方向一般被认为始自 1994 年美国明尼苏达大学 GroupLens 研究组推出的 GroupLens 系统（Resnick,1994），如图 6.2 所示。该系统有两大重要贡献：一是首次提出了基于协同过滤（Collaborative Filtering，CF）来完成推荐任务的思想，二是为推荐问题建立了一个形式化的模型（下述）。基于该模型的协同过滤推荐引领了之后推荐系统十几年的发展方向。

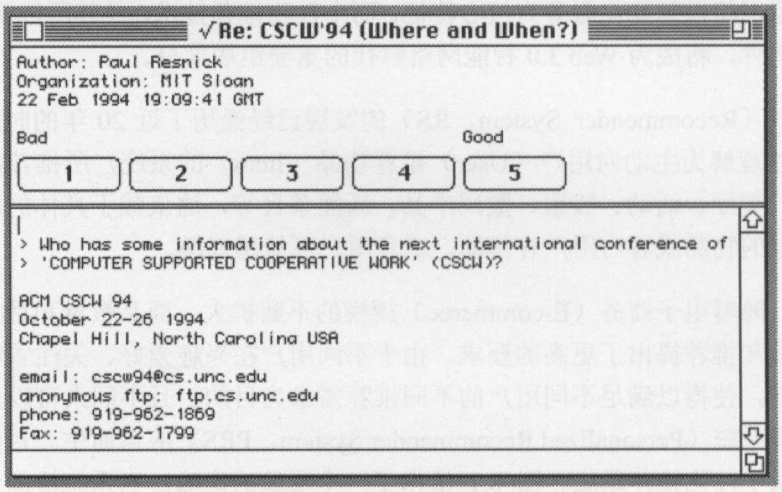


图 6.2 Resnick 推出的 GroupLens 系统

GroupLens 所提出的推荐算法实际上就是目前人们时常提及的基于用户的协同过滤推荐算法（User-based Collaborative Filtering，User-based CF），虽然其论文本身并没有使用这样一个名字。在之后的十几年中，其他一些著名的协同过滤算法相继被提出，主要的有基于物品的协同过滤算法（Item-based Collaborative Filtering，Item-based CF）（Sarwar,2001）、基于矩阵分解的协同过滤算法（Matrix Factorization-based Collaborative Filtering，MF-based CF），等等。在 2006 年举办的 Netflix 大奖赛中，核心任务之一为预测用户对电影的评分，而在这一赛事中，获奖团队所采用的方法，其核心模块即为矩阵分解，这也使基于矩阵分解的协同过滤在接下来的近十年中得到了广泛的关注和重视，成为协同过滤的主流范式之一。当然，基于其它方法而非协同过滤的推荐算法也在不断地发展，如基于内容的推荐（Content-based Recommendation）、排序学习（Learning to Rank），以及借助情感分析、主题模型的基于文本的推荐（Review-based Recommendation）等。另外，这些方法之间的互补、

融合也成为重要的研究方向。

在本章，我们将介绍个性化推荐的应用、基本问题、主要方法、基本数学原理和发展前景，并重点讲述个性化推荐的基本范式之一——矩阵分解算法。读者只需要拥有简单的矩阵运算基础知识。

6.1.2 推荐无处不在

目前，推荐系统几乎成为了绝大多数网络应用的必需模块，广泛存在于各种互联网产品中，如图 6.3 所示。只要涉及到我们希望根据用户的喜好、让不同用户获得不同的展示结果的地方，背后就往往需要有个性化推荐技术的支持。

图 6.3 无处不在的个性化推荐

当我们在优酷、土豆等视频网站观看视频时，会看到系统推荐的其他相关视频；在豆瓣 FM 上收听音乐时，可以获得高度个性化的音乐推荐；在淘宝、天猫、京东等购物网站上购物时，会看到系统推荐的其它可能需要的商品；在微博、人人网等社交网站上浏览网页时，则可以获得话题推荐、好友推荐等各种形式的个性化推荐；更有频繁出现在各种网络应用中各式各样的互联网广告，也是个性化推荐的重要战场，为企业带来了丰厚的收益。据著名的全球网络零售网站亚马逊发布的数据显示，亚马逊网络书城的推荐算法为亚马逊每年贡献近三十个百分点的创收，因此推荐系统对互联网企业收入的重要性可见一斑。

6.1.3 从千人一面到千人千面

个性化推荐技术的核心在于“个性化”（Personalize），而“推荐”（Recommendation）只是“个性化”下的一种应用场景，除此之外，我们还可以构建个性化搜索引擎，个性化

手机助手等基于个性化技术的应用。一言以蔽之，个性化技术所要解决的核心问题，就是在描述用户上实现从“千人一面”到“千人千面”的技术升级。

为了直观地了解个性化的效用，我们以“传统的”（非个性化的）搜索为例。在非个性化的搜索引擎中，用户输入查询语句（Query），系统则给出该查询的结果。在最简单的设置下，不同用户，只要其输入的查询语句是一样的，得到的搜索结果也将是一样的，与不同用户的兴趣、喜好、历史行为等信息无关。在这一设置下，系统能了解用户的途径只有用户输入的查询语句，也就是说，这条语句就成为算法刻画电脑背后的 TA 的唯一信息，正是因为（拥有不同查询目的和需求的）不同用户所使用的查询语句有可能是一样的，系统也就没有办法实现个性化（对用户做区分）并给出个性化的查询结果（不同用户在同一查询下获得的结果不同）。

融入了个性化因素的个性化搜索引擎（Personalized Search）则突破这一限制，试图在同一查询下为用户提供个性化的、不同的检索结果。然而我们为什么一定要试图为不同用户提供不同的结果呢？举例而言，同样是输入“苹果”（Apple）作为查询语句的用户，其期望得到的结果有可能是苹果这种水果，也有可能是苹果手机、电脑等电子产品，亦或有可能是《苹果》这部电影。同样的查询语句，用户背后真正的信息需求有可能是不一样的，如果我们能够识别出用户个性化的信息需求，从而更有可能将符合其需求的结果排在查询列表的前面，就可以更好地提升用户体验。

而对用户不同信息需求的识别，就需要用到用户的个性化建模和个性化搜索技术、算法。这里的个性化建模可以来自用户的注册信息（如水果经销商、电子产品经销商，或影评人），也有可能来自用户的历史搜索信息。在最近，百度、谷歌等搜索引擎公司都已建立了较为完善的账号体系，只要用户登录了个人账户，就可以更好地利用用户全方位的个性化信息、为用户提供更方便的个性化检索服务。这种服务超越了以往简单使用查询语句对用户进行描述的方法，而是使用更多的信息对用户进行全方位的个性化描述——这恰恰就是从千人一面走向千人千面的过程。

6.2 个性化推荐的基本问题

在对个性化推荐的具体理论算法进行数学化的介绍之前，我们先对个性化推荐的基本问题进行介绍，并对其框架进行形式化，从而方便读者的理解。在本节，我们将由浅入深依次介绍推荐系统的输入/输出、个性化推荐的形式化、以及推荐系统所需要解决的三大核心问题——预测、推荐和解释。

6.2.1 推荐系统的输入

推荐系统可能的输入数据及其形式多种多样，传统的推荐算法，其输入归纳起来可以分为用户（User）、物品（Item）和评价（Review）三个方面，它们分别对应于一个矩阵中的行、列、值。

其中“物品”用来描述一个对象的性质，也经常被称为物品属性（Item Profile）。需要注意的是，这里的“物品”概念非常广泛，可以是用户在互联网上有可能面对的任何对象，比如网络新闻、视频、音乐、电子图书、广告、社交网站中的好友等，而不仅仅是购物网站中的商品。根据物品的不同，其属性当然也不尽相同。比如对于图书推荐，物品属性有可能包括图书所属类别、作者、页数、出版时间、出版商等；而对于新闻推荐，物品的属性则有可能是新闻的文本内容、关键词、时间等；对于电影，可以是片名、时长、上映时间、主演、剧情描述，等等。

这里的“用户”不仅仅可以是一个用户的 ID，还可以是用来描述一个用户个性的“用户画像”（User Profile）。根据不同的应用场景以及不同的具体算法，用户画像可能有不同的表示形式。一种直观且容易理解的形式是用户的注册信息，比如该用户的性别、年龄、年收入、活跃时间、所在城市等。物品和用户的关系可以用图 6.4 来形象地表示。

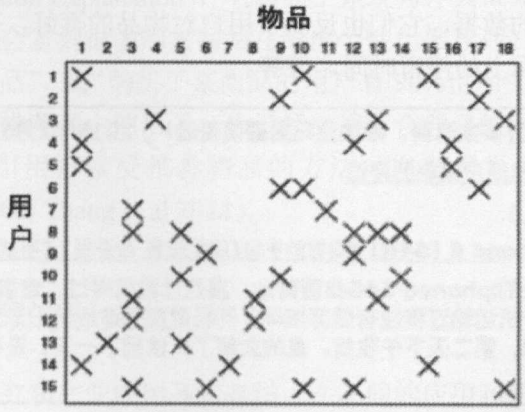


图 6.4 物品和用户的关系

但是在推荐系统中，这样的画像很难集成到常见的算法中，也很难与具体的物品之间建立联系。比如我们很难断定某商品一定不会被某年龄段的人喜欢，因为这样的判断过于粗糙。因此这种画像在推荐系统中虽然也经常会被使用，但是很少直接用在推荐算法中，而是用于对推荐结果进行过滤和排序。

由于在很多推荐算法中，计算用户画像和物品属性之间的相似度是一个经常会用到的操作，故另一种使用更为广泛也更有实际意义的用户画像应运而生（Sugiyama, et al. 2004）。*它的结构与该系统中物品属性的结构一样，为了更清楚地说明其结构，我们以一种典型的*构建用户画像的方法为例来进行说明：考虑该用户进行过浏览或评分的所有物品，将这些物品在每一项属性上获得的打分分别进行加权平均，得到一个综合的属性，作为该用户的画像。这种用户画像的优点是非常容易计算其与物品之间的相似度，同时比较准确地描述了该用户在物品上的偏好，巧妙地避开了用户私人信息这一很难获得的数据，具有保护隐私的能力；进一步，如果加入时间因素，还可以研究用户在物品上偏好的变化等，因此得到广泛应用。

评价（Review）是联系一个用户与一个物品的纽带，最简单也是最常见的评价是购物网站中用户对某一物品的打分（Rating），如图 6.5 所示是一条来自亚马逊购物网站的用户评论，其中的五星评分体系经常被各大电子商务网站采用，它表示该用户对该物品的喜好程度，而在常见的推荐算法中，它被描述为一个 1～5 的整数。当然，用户对物品或信息的偏好，根据应用本身的不同，还可能包含很多不同的信息，比如用户对商品的评论文本（Review Text）、用户查看的历史记录、用户购买的记录等，这些信息总体上可以分为两类：一是显式的用户反馈（Explicit Feedback），这是用户对商品或信息给出的显式反馈信息，评分、评论属于该类；另一类是隐式的用户反馈（Implicit Feedback），这类一般是用户在使用网站的过程中产生的数据，它们也反映了用户对物品的喜好，比如用户查看了某物品的信息，用户在某一页面上的停留时间，等等。

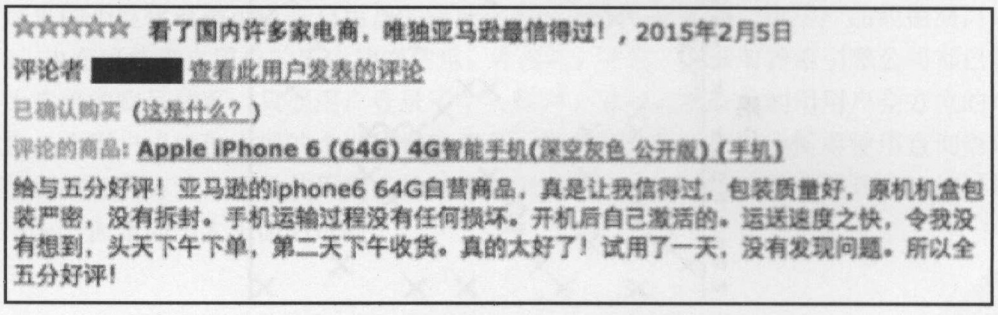


图 6.5 亚马逊购物网站的一条购物评论

虽然目前大多数的推荐算法往往都基于用户评分矩阵（Rating Matrix），但是基于用户评论、用户隐式反馈数据的方法来完成推荐越来越受到人们的关注，这些方面的研究长期以来受到文本挖掘、用户数据收集等方面的难点的制约，没有得到充分的研究，但是它们在解决推荐系统的可解释性、冷启动问题等方面确实具有重要的潜力（Wietsma & Ricci 2005; Ricci & Wietsma 2006; Aciar, et al. 2007; McAuley 2013; Zhang et al 2014）。

6.2.2 推荐系统的输出

对于一个特定的用户，推荐系统的输出一般是一个个性化的“推荐列表”（Recommendation List），如图 6.6 所示，该推荐列表按照优先级的顺序给出了对该用户可能感兴趣的物品。

图 6.6 个性化的“推荐列表”

对于一个实用的推荐系统而言，仅仅给出推荐列表往往是不够的，因为用户不知道为什么系统给出的推荐是合理的。如果用户对系统给出的推荐结果不满意，也不能理解为何会给出这样的推荐结果，则很难促使用户采纳系统给出的推荐，甚至会极大地伤害用户使用推荐系统乃至整个系统的体验。为了解决这个问题，推荐系统另一个重要的输出是“推荐理由”（Recommendation Explanation），它描述了系统为什么认为推荐该物品是合理的。如果读者稍加注意，就会看到很多购物网站在给出个性化推荐列表的同时会给出“根据您的浏览历史推荐如下商品”，或“购买了某商品的用户有 90%也购买了该商品”等推荐理由，这就是我们经常见到的推荐理由的形式。为了解决推荐合理性的问题，推荐理由在产业界被作为一个重要的吸引用户接受推荐物品的方法，在学术界也受到越来越多的关注（Tintarev & Masthoff 2000; Zhang et al 2014）。

6.2.3 个性化推荐的形式化

推荐系统的学术研究和产业应用各式各样，在不同的应用和研究背景下，对个性化推荐问题的形式化也多种多样。在这里，为了方便读者较为深入地了解推荐系统的基本问题和方法，我们给出个性化推荐问题一个最典型、最常用、也是从最初至今一直被沿用了几十年的形式化方法。如上所述，该形式化方法最早来自于 GroupLens 研究组（Resnick & Iacovou et al. 1994），并在（Herlocker et al. 1999）中做了进一步的阐述。

首先我们拥有一个大型的矩阵，该矩阵的每一行表示一个用户，每一列表示一个物品，矩阵中的每一个数值表示该用户对该物品的打分，这经常是一个 1～5 的分值，对应于用户

在购物网站中对商品给出的一星到五星的评价，“1”表示该用户对该商品最不满意，“5”则表示该用户对该物品非常满意，如果某用户对某物品没有评分，则对应的矩阵元素值为“0”。视我们所拥有数据的具体情况，每一个用户可能有其对应的用户画像，而每一个物品，也可能有其对应的物品属性。

需要指出的是，这样一个用户-物品评分矩阵往往非常稀疏（Sparse），也就是说该矩阵中往往有大量的“0”值，而只有少量的非零值。这是因为相对于一个系统（如购物网站）中数量庞大的物品而言（如购物网站中的全部商品），一个用户个体所真正浏览或购买过的物品往往非常少。举例而言，在著名的餐厅评论网站 Yelp 的用户-物品行为矩阵中，有近一半（约 49%）的用户只有一个评分行为，矩阵的稀疏度（0 的个数在矩阵中所占的百分比）更是达到 99.96%。由此可见，在真实的系统中，我们所能利用的用户行为数据，相比系统中未知的信息而言少之又少，这就是个性化推荐的一个难点所在。

我们现在解决这样一个问题：给定如上的稀疏矩阵之后，对于某一个用户，向其推荐哪些他没有打过分的物品最容易被他接受，这里的“接受”根据具体的应用环境有所不同，有可能是查看该新闻、购买该商品、收藏该网页，等等。对于推荐算法，还需要一系列的评价指标来评价推荐的效果，这些评价方法和评价指标将在后面的部分具体说明。

6.2.4 推荐系统的三大核心问题

有了如上的形式化描述之后，推荐系统所要解决的核心问题主要有三个，分别是预测（Prediction）、推荐（Recommendation）和解释（Explanation）。

“预测”模块所要解决的主要问题是推断每一个用户对每一个物品的喜好程度，长期以来，其主要手段是根据如上稀疏矩阵中已有的信息（打分或评论）来计算用户在他没打过分的物品上可能的打分或喜好程度。

“推荐”模块所要解决的主要问题则是根据预测环节所计算的结果向用户推荐他没有打过分的物品。由于物品的数量众多，用户不可能全部浏览一遍，因此“推荐”的核心步骤是对推荐结果的排序（Ranking）。当然，按照预测分值的高低直接排序确实是一种比较合理的方法，但是在实际系统中，排序往往要考虑更多更复杂的因素，比如用户的年龄段、用户在最近一段时间内的购买记录等，因而我们对用户画像的结果往往也在这个环节派上用场。

“解释”模块则对如上所给出的推荐列表中的每一个物品或推荐列表整体给出解释，即为何我们认为这样的一个推荐列表对用户而言是合理的，从而说服用户去查看甚至接受我们给出的推荐。这样的解释可以以各种可能的形式出现，而不仅限于一句解释性的语言。

例如通过词云描述被推荐物品的主要属性，从而帮助用户一目了然地理解被推荐物品与自己个性化需求之间的相似之处；甚至透过关系图谱展示被推荐物品与用户已购买物品的关系，等等。

虽然人们早就意识到预测、推荐和解释作为推荐系统的三大核心模块都具有重要的作用，但是目前绝大多数的推荐算法都仍然把精力集中在“预测”环节上，并提出了基于内容的方法、基于协同过滤的方法、尤其是各类理论基础和实际效果都比较扎实的矩阵分解算法，等等。而推荐和解释作为重要的后续环节需要更多的研究与探索，这与搜索引擎的发展非常类似。除此之外，推荐多样性（Ziegler, et al. 2005）、推荐系统的界面等很多方面的问题也在受到越来越多的关注。

6.3 典型推荐算法浅析

在本节，我们对目前常见的推荐算法进行归类整理，分析它们的共同之处和不同点，并对常见的推荐方法进行介绍。本节力图使读者对各式各样的个性化推荐算法及其之间的关系有一个整体的认识，并了解典型方法的特点。

6.3.1 推荐算法的分类

按照不同的分类指标，推荐系统具有很多不同的分类方法，常见的分类方法有依据推荐结果是否因人而异、依据推荐方法的不同、依据推荐模型构建方式的不同等。

依据推荐结果是否因人而异，主要可以分为大众化推荐和个性化推荐。大众化推荐往往与用户本身及其历史信息无关，在同样的外部条件下，不同用户获得的推荐是一样的。大众化推荐一个典型的例子是查询推荐，它往往只与当前的 query 有关，而很少与该用户直接相关。个性化推荐的特点则是不同的人在同样的外部条件下，也可以获得与其本身兴趣爱好、历史记录等相匹配的推荐。例如，在目前的搜索引擎技术的研究中也越来越引入个性化方法，使得查询推荐不仅与当前的查询语句有关，也与当前用户的个性化信息有关。

依据推荐方法的不同，推荐算法大致可以分为如下几种：基于人口统计学的推荐（Demographic-based Recommendation）（Pazzani 1999）、基于内容的推荐（Content-Based Recommendation）（Gunawardana & Shani 2009）、基于协同过滤的推荐（Collaborative Filtering-Based Recommendation），以及混合型推荐系统（Hybrid Recommendation）（Melville,

et al. 2002)。其中基于协同过滤的推荐被研究人员研究得最多也最为深入，它又可以被分为多个子类别，包括基于用户的推荐（User-Based Recommendation）(Resnick, et al. 1994)，基于物品的推荐（Item-Based Recommendation）(Sarwar, et al. 2001)，基于社交网络关系的推荐（Social-Based Recommendation）(Kautz, et al. 1997)，基于模型的推荐（Model-based Recommendation)，等等。其中，基于模型的推荐是指利用系统已有的数据，学习和构建一个模型，进而利用该模型进行推荐，这里的模型可以是SVD、NMF等矩阵分解模型(Sarwar, et al. 2000A)，也可以是利用贝叶斯分类器、决策树、人工神经网络等模型转化为分类问题，或者基于聚类技术对数据进行预处理的结果（George & Merugu 2005)，等等。

而依据推荐模型构建方式的不同，目前的推荐算法大致可分为基于用户或物品本身的启发式推荐（Heuristic-Based，或称为 Memory-Based Recommendation)、基于关联规则的推荐（Association Rule Mining for Recommendation）(Sarwar, et al. 2007)，基于模型的推荐（Model-based Recommendation)，以及混合型推荐系统（Hybrid Recommendation)。

6.3.2 典型推荐算法介绍

1. 基于人口统计学的推荐

基于人口统计学的推荐（Demographic-Based Recommendation）虽然已经很少被单独使用，但是理解这种方法的工作原理对于深入理解推荐系统有很大帮助。基于人口统计学的方法基于假设“一个用户有可能会喜欢与其相似的用户所喜欢的物品”。它首先记录了每一个用户的性别、年龄、活跃时间等元数据，当我们需要对一个用户进行个性化推荐时，利用其元数据计算他与其他用户之间的相似度，并选出与其最相似的一个或几个用户，进而利用这些用户的购买和打分历史记录进行推荐。一种简单且常见的推荐方法就是将这些(最相似的）用户所覆盖的物品作为推荐列表，并以物品在这些用户上的平均得分作为依据来进行排序。

基于人口统计学的推荐方法其优点是计算简单，由于用户的元数据相对比较稳定，因此相似用户的计算可以线下完成，从而便于实现实时响应。但它同时也有诸多问题，其主要问题之一便是计算可信度较低，因为即便是性别、年龄等元数据属性都相同的用户，也很有可能在物品上有截然不同的偏好，因此这种计算用户相似度的方法往往并不能与物品之间建立真正可靠的联系。 因此，基于人口统计学的方法在实际推荐系统中很少作为一个特定的方法单独使用，而常常与其他方法结合，利用用户元数据对推荐结果进行进一步地优化。

2. 基于内容的推荐

基于内容的推荐（Content-Based Recommendation）则假设“一个用户可能会喜欢和他曾经喜欢过的物品相似的物品”，而这里“相似的物品”则通过商品的内容属性来确定，例如电影的主要演员、风格、时尚，音乐的曲风、歌手，商品的价格、种类，等等。典型的基于内容的方法首先需要构建用户的画像（Profile），一种较为简单的方法是考虑该用户曾经购买或浏览过的所有物品，并将这些物品的内容信息加权整合作为对应用户的画像，它描述了一个用户对物品属性的偏好特征。当然，构建用户画像的策略可以很复杂，比如可以考虑时间因素，计算用户在不同时间段内的画像，从而了解该用户在历史数据上所表现出来的偏好变化，等等。有了用户画像之后，我们就可以开始推荐了，最简单的推荐策略就是计算所有该用户未尝试过的物品与该用户的画像之间的相似度，并按照相似度由大到小的顺序生成推荐列表，作为推荐结果。当然，推荐策略也可以很复杂，比如在数据源上考虑本次用户交互过程中所收集到的即时交互数据来决定排序，在模型上使用决策树、人工神经网络，等等。但是这些方法最核心的环节都是利用用户画像和物品属性之间的相似度计算。

其实在很多基于内容的推荐算法中，并不把用户画像显式地计算出来，而是利用用户打过分的物品，直接计算推荐列表。一种直观的方法就是计算它与该用户尝试过的所有物品之间的相似度，并将这些相似度根据用户的打分进行加权平均。这实际上也是基于内容的方法，只是绕过了计算用户画像的环节。实际上，很多具体的应用表明，绕过用户画像的计算，转而直接利用商品属性计算相似度，往往更灵活，并获得更好的推荐效果，这是因为在计算用户画像的过程中，一些有用的信息被丢掉以至于无法在后面的环节中被利用。

基于内容的推荐方法对于解决新物品的冷启动（Cold-Start）问题有重要的帮助，这是因为只要系统拥有该物品的属性信息，就可以直接计算它与其他物品之间的关联度，而不受用户评分数据稀疏性的限制；另外，推荐结果也具有较好的可解释性，一种显然的推荐理由是“该物品与用户之前曾经喜欢过的某物品相似”。然而基于内容的推荐方法也有一些缺点，首先是系统需要复杂的模块甚至手工来预处理物品信息以得到能够代表它们的特征，然后受信息获取技术的制约、处理对象的复杂性高等因素，这样工作难以达到较好的效果；其次，该方法难以发现用户并不熟悉但是具有潜在兴趣的物品，因为该方法总是倾向于向用户推荐与其历史数据相似的物品；另外，该方法往往不具有较好的可扩展性，需要针对不同的领域构建几乎完全不同的物品属性，因而针对一个数据集合训练的模型未必适合其他的数据集合。

3. 基于协同过滤的推荐

基于协同过滤的推荐（Collaborative Filtering-Based Recommendation）一般指通过收集

用户的历史行为和偏好信息，并利用群体的智慧（Wisdom of the Crowds）为当前用户给出个性化的推荐。根据前文所述，基于协同过滤的推荐大致包括基于用户的推荐（User-based Recommendation），基于物品的推荐（Item-based Recommendation）和基于模型的推荐（Model-based Recommendation），等等。

基于用户的推荐方法是最早的一种基于协同过滤的推荐算法（Resnick, et al. 1994），其基本假设与基于人口统计学的方法类似，即“用户可能会喜欢和他具有相似爱好的用户所喜欢的物品”，然而它们重要的不同之处在于，这里的“相似用户”不是用用户的人口统计信息直接计算出来的，而是利用用户的打分历史记录来进行计算的。其基本的假设为具有相似偏好的用户在物品上的打分情况往往具有更强的相似性。

基于用户的推荐方法其核心在于最近邻搜索，我们把每一个用户看成一个行向量，并计算其他用户行向量与该用户的相似度，而这里的相似度计算可以采用多种不同的指标，如 Pearson 相关性系数、余弦相似度等。当我们拥有了用户之间的两两相似度之后，选择与目标用户最相似的前 k 个用户的历史购买/浏览行为信息为目标用户给出个性化的推荐列表。例如在 Top-N 推荐中，系统统计在这前 k 个用户中出现频率最高且在目标用户的历史记录中未出现的物品，从而利用这些物品构建推荐列表作为输出。关联推荐的基本思想则是利用这前 k 个用户的购买或打分记录进行关联规则挖掘，并利用挖掘出的关联规则结合目标用户的购买记录完成推荐，典型的推荐结果如很多网络购物商城中常见的“购买了某物品的用户还购买了该物品”。

在基于用户的推荐方法中，“个性化”体现在对于不同的用户而言其最近邻是不同的，从而得到的推荐列表也不尽相同；“协同过滤”则体现在对目标用户进行推荐时使用了其他用户在物品上的历史行为信息，这是与基于人口统计学的推荐方法的不同之处。

基于用户的方法优点在于在数据集完善、内容丰富的条件下能够获得较高的准确率，而且能够对物品的关联性和用户的偏好进行隐式透明的挖掘。而其缺点则在于随着系统用户数量的增大计算用户相似度的时间代价会显著增长，使得该方法难以胜任用户量变化巨大的系统，从而限制了算法的可扩展性。另外，冷启动用户的问题也是该方法难以处理的重要问题：当新用户加入系统时，由于其打分历史记录很少，难以准确计算该用户的相似用户，这也进一步引出数据稀疏性对系统可扩展性的限制。

鉴于基于用户的协同过滤方法可扩展性差的问题，研究人员进一步提出了基于物品的推荐（Item-based Recommendation）（Sarwar, et al. 2001）。基于物品的推荐方法所基于的基本假设与基于内容的方法类似，也就是“用户可能会喜欢与他之前曾经喜欢的物品相似的物品”。比如喜欢《长尾理论》这本书的人，也很有可能去看《世界是平的》，然而与基于

内容的推荐方法不同的是，这里的“相似物品”并非通过物品属性来计算，而是通过网络用户对物品的历史评分记录来计算的。

基于物品的推荐方法将矩阵的每一个列向量作为一个物品来计算物品列向量之间的相似度，并基于物品之间的两两相似度进行预测和推荐。一个简单的例子是在当用户购买了某一商品后直接向其推荐与该物品相似度最高的前几个商品；或者更为复杂一点，考虑该用户所有的历史打分记录（Sarwar, et al. 2001），并对一个用户行向量中的 0 值（即用户未购买的物品）预测用户在该物品上可能的打分，如图 6.7 所示。例如，我们可以考虑目标用户在历史上所有打过分的物品，并以它们与待预测的物品的相似度为权重对这些历史打分值进行加权平均，作为对待预测的目标物品的预测打分，最终以预测打分的高低为顺序给出推荐列表。基于物品的推荐方法总体上来说是一种启发式方法，对目标的拟合能力是有限的，但是当把多个启发式方法结合起来，也可以有很好的拟合能力（Sarwar, et al. 2001）。

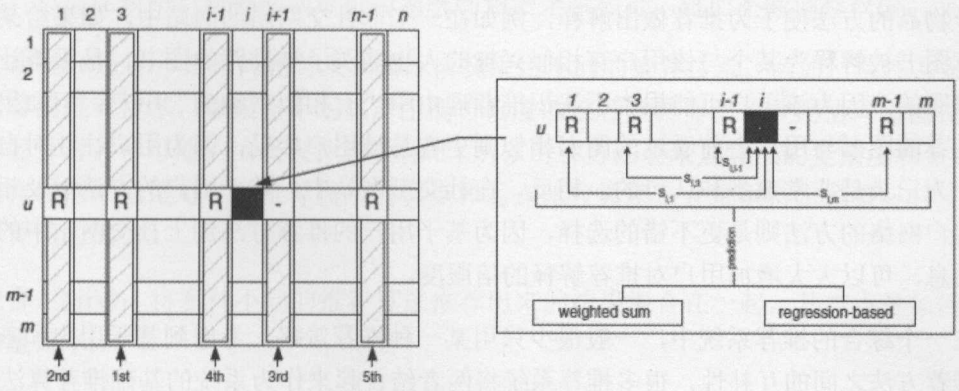


图 6.7 基于物品的推荐方法矩阵

基于物品的推荐方法其优点主要有以下两点：一是计算简单，容易实现实时响应，由于在常见的系统中物品被评分的变化要比用户低得多，因此物品相似度的计算一般可以采用离线完成、定期更新的方式，从而减少了线上计算，实现实时响应并提高效率，尤其是在用户数远大于商品数的情况下效果更为显著，例如用户新添加了几个感兴趣的商品之后就可以立即给出新的推荐；二是可解释性较好，用户可能不了解其他人的购物情况，但是对自己的购物历史总是很清楚的；另外用户总是希望自己有最后的决定权，如果对系统推荐的商品不满意，需要有办法让用户改进它，基于物品的推荐方法很容易让用户理解为什么推荐了某个商品，并且当用户在兴趣列表里添加或删除商品时，可以调整系统的推荐结果，这也是其他方法最难做到的一点。然而基于物品的推荐也有其缺点：以物品为基础的信息过滤系统较少考虑用户之间的差别，因此精度较基于用户的方法往往稍微逊色一些；除此之外，还有许多其他的问题有待解决，最典型的就是数据稀疏性和冷启动的问题。

由于基于用户的推荐和基于物品的推荐具有某种对称性，且均为个性化推荐系统最为基础的入门级方法，因此在这里我们对这两种最基本的协同过滤方法进行对比。

在计算复杂性上，基于用户的方法往往在线计算量大，难以实时响应。对于一个用户数量大大超过物品数量而物品数量相对稳定的应用，一般而言基于物品的方法从性能和复杂度上都比基于用户的方法更优，这是因为物品相似度的计算不但计算量较小，而且不必频繁更新；而对于诸如新闻、博客或者微内容等物品数量巨大且更新频繁的应用，基于用户的方法往往更具优势，推荐系统的设计者需要根据自己应用的特点选择更加合适的算法。

在适用场景上，内容之间的内在联系是非社交型网站中很重要的推荐原则，往往比基于相似用户的推荐原则更加有效。例如在购书网站上，当用户看一本书的时候，推荐引擎会给用户推荐与其相关的书籍，这个推荐的重要性远远超过了网站首页对该用户的综合推荐。可以看到，在这种情况下，基于物品的推荐方法成为了引导用户浏览的重要手段。同时基于物品的方法便于为推荐做出解释，例如在一个非社交网络的网站中，如果给某个用户推荐图书被解释为某个与该用户有相似兴趣的人也购买了被推荐的图书，是很难让目标用户信服的，因为该用户可能根本不认识推荐理由中“有相似兴趣的”用户；但如果解释为被推荐的图书与用户之前看过的图书相似则更容易被用户接受，因为用户往往对自己的历史行为记录是非常熟悉和认可的。相反，在社交型网站中，基于用户的方法以及相关的基于用户网络的方法则是更不错的选择，因为基于用户的推荐方法加上社交网站中的社会网络信息，可以大大增加用户对推荐解释的信服度。

在一个综合的推荐系统中，一般很少只用某一种推荐策略，考虑到基于用户和基于物品的推荐方法之间的互补性，很多推荐系统将两者结合起来作为系统的基础推荐算法。

4. 基于模型的推荐（Model-Based Recommendation）

基于用户或基于物品的方法共有的缺点是计算规模庞大，并难以处理大数据量下的实时结果。以模型为基础的协同过滤技术则致力于改进该问题：我们首先利用历史数据训练得到一个模型，再用此模型进行预测。

以模型为基础的协同过滤广泛使用的技术包括潜在语义分析（Latent Semantic Analysis）、贝叶斯网络（Bayesian Networks）、矩阵分解（Matrix Factorization），等等。它们收集用户的打分数据进行分析和学习并推断出用户行为模型，进而对某个产品进行预测打分。例如可以将用户属性和物品属性中的各个特征作为输入，以用户打分作为输出来拟合回归模型；或者将打分作为类别转化为一个多分类器问题，等等。这种方式不是基于一些启发规则进行预测计算，而是对于已有数据应用统计和机器学习得到的模型进行预测。

基于模型的方法的优点在于快速响应，即只要训练出了模型就可以对新用户或新物品进行实时快速计算，并且由于可以直接以用户打分作为优化目标，往往可以获得较高的预测精度。然而其问题在于如何将用户实时或者新增的喜好信息反馈给训练好的模型，从而在系统扩展的过程中维持甚至提高推荐的准确度，也就是模型的增量训练问题。

5. 混合型推荐方法（Hybrid Recommendation）

混合型推荐系统和算法是推荐系统的另一个研究热点，它是指将多种推荐技术进行混合相互弥补缺点来获得更好的推荐效果。最常见的是将协同过滤技术和其他技术相结合以克服冷启动的问题。常见的混合型推荐策略有如下几种（Burke 2002）。

加权融合（Weighted）：将多种推荐技术的计算结果加权混合产生推荐，最简单的方式是基于感知器的线性混合，首先将协同过滤的推荐结果和基于内容的推荐结果赋予相同的权重值，然后比较用户对物品的评价与系统的预测是否相符，进而不断调整权值。

切换（Switch）：根据问题背景和实际情况采用不同的推荐技术。例如，系统首先使用基于内容的推荐技术，如果它不足以产生高可信度的推荐就转而尝试使用协同过滤技术。因为需要针对各种可能的情况设计转换标准，所以这种方法会增加算法的复杂度和参数化，当然这样做的好处是对各种推荐技术的优点和弱点比较灵敏，并根据特定场景充分发挥不同推荐算法的优势。

混合（Mix）：将多种不同的推荐算法推荐出来的结果混合在一起，其难点是如何进行结果的重排序。

特征组合（Feature Combination）：将来自不同推荐数据源的特征组合起来，由另一种推荐技术采用。这种方法一般会将协同过滤的信息作为增加的特征向量，然后在这增加的数据集上采用基于内容的推荐技术。特征组合的混合方式使得系统不再仅仅考虑协同过滤的数据源，所以它降低了用户对物品评分数量的敏感度。相反，它允许系统拥有物品的内部相似信息，对协同系统是不透明的。

级联型（Cascade）：用后一个推荐方法优化前一个推荐方法。它是一个分阶段的过程，首先用一种推荐技术产生一个较为粗略的候选结果，在此基础上使用第二种推荐技术对其作出进一步精确的推荐。

特征递增（Feature Augmentation）：将前一个推荐方法的输出作为后一个推荐方法的输入，它与级联型的不同之处在于，这种方法上一级产生的并不是直接的推荐结果，而是为下一级的推荐提供某些特征。一个典型的例子是将聚类分析环节作为关联规则挖掘环节的预处理，从而将聚类所提供的类别特征用于关联规则挖掘。

元层次混合（Meta-level hybrid）：将不同的推荐模型在模型层面上进行深度的融合，而不仅仅是把一个输出结果作为另一个的输入。例如，基于用户的方法和基于物品的方法一种可能的组合方式为：先计算目标物品的相似物品集，然后删掉所有其他（不相似的）物品，进而在目标物品的相似物品集上采用基于用户的协同过滤算法。这种基于相似物品计算近邻用户的协同推荐方法，能很好地处理用户多兴趣下的个性化推荐问题，尤其是在候选推荐物品的内容属性相差很大的时候，该方法可以获得较好的性能。

6.3.3 基于矩阵分解的打分预测

1. 矩阵上的打分预测问题及其评价

在前面的介绍中我们已经知道，一个典型的推荐系统常常把用户和物品之间的关系形式化为一个稀疏矩阵（如图6.8所示的示例），其中矩阵的每一行对应一个用户，每一列对应一个物品，矩阵中的每一个非零值（图中以“×”标记的元素）代表相应的用户对物品的打分（一般是1～5的星级打分），而每一个零值（图中空白的部分）则代表用户在历史上没有对该物品进行过评分。在这样一个矩阵上的打分预测问题即为根据矩阵中已有的值预测矩阵中缺失的值，也就是尽可能精确地估计一个用户在未买过的物品上可能的打分，从而基于预测打分的高低给出推荐列表。

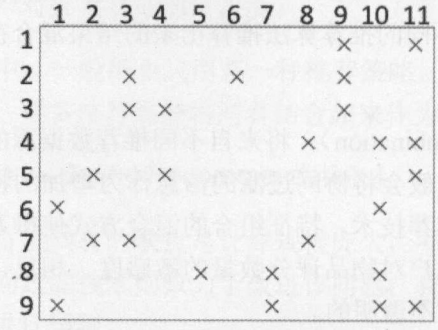


图6.8 用户和物品之间的稀疏矩阵关系

为了设计好的打分预测算法，我们首先需要定义合适的评价指标来评价一个算法的预测结果，常用的评价指标为根分均方差（Root Mean Square Error, RMSE）和平均绝对误差（Mean Absolute Error, MAE）。设矩阵以 X 来表示，矩阵中的每一个打分记为 r_{ij}，所有打分的集合记为 S，我们一般取一部分的打分（如80%）来进行模型的训练和验证，并用其他的部分（如20%）来进行评价。设 $\hat{r}_{ij}$ 表示我们算法所给出的预测打分，并以 $\hat{S}$ 表示所有用于测试的打分值集合，那么评价指标RMSE和MAE的计算如下所示：

$$RMSE=\sqrt{\frac{\sum_{r_{ij}\in\hat{S}}(r_{ij}-\hat{r}_{ij})^2}{|\hat{S}|}},MAE=\frac{\sum_{r_{ij}\in\hat{S}}|r_{ij}-\hat{r}_{ij}|}{|\hat{S}|}$$

一个评分预测算法致力于预测矩阵中未知的打分，并使得 RMSE 或 MAE 评价指标最小。在接下来的部分中，我们介绍在推荐系统中广泛使用的基于矩阵分解的预测算法及其统一的形式化表示。

2. 矩阵分解算法的形式化

基于矩阵分解的矩阵补全由于其较好的预测精度和较高的可扩展性，故在实际推荐系统中得到了广泛的应用。目前，研究人员设计了诸多基于矩阵分解的矩阵补全和预测算法，例如矩阵的奇异值分解（Singular Value Decomposition，SVD）、非负矩阵分解（Non-negative Matrix Factorization, NMF）、概率化矩阵分解（Probabilistic Matrix Factorization, PMF）、最大间隔矩阵分解（Maximum Margin Matrix Factorization, MMMF），等等。例如图 6.9 的示例展示了使用非负矩阵分解对原始矩阵的未知值进行预测的结果。

	I1	I2	I3	I4
U1	5	3	-	1
U2	4	-	-	1
U3	1	1	-	5
U4	1	-	-	4
U5	-	1	5	4

⇒

	I1	I2	I3	I4
U1	4.97	2.98	2.18	0.98
U2	3.97	2.40	1.97	0.99
U3	1.02	0.93	5.32	4.93
U4	1.00	0.85	4.59	3.93
U5	1.36	1.07	4.89	4.12

图 6.9 使用非负矩阵分解算法预测原始矩阵的未知值

但是，大部分已有的矩阵分解算法都可以用一个统一的模型进行概括。为了帮助读者更好地了解矩阵分解算法的本质以及不同算法之间深刻的内在联系，我们首先介绍矩阵分解算法一种常用的统一表示形式。

设 $X\in R^{m\times n}$ 是一个稀疏矩阵，并设 $U\in R^{m\times r}$ 和 $V\in R^{n\times r}$ 是对原始矩阵 X 的低秩分解，那么一个矩阵分解算法 $P=(f,D_W,C,R)$ 可以形式化概括为如下各部分的组合：

预测函数 f：$R^{m\times n}\rightarrow R^{m\times n}$

可选的权重矩阵 $W\in R_+^{m\times n}$，当权重矩阵存在时，它往往是损失函数的一部分。

损失函数 $D_W(X,f(UV^T))\geqslant 0$，它表示当我们用预测矩阵 $f(UV^T)$ 来近似 X 时所引入的预测误差。

对分解矩阵的某种约束条件 C：$(U,V)\in C$

正则化因子：$R(U,V)\geqslant 0$

在如上的各部分下，原始矩阵 X 被近似为 $\hat{X}=f(UV^T)$，同时一个矩阵分解算法可以形式化为如下的最优化问题：

$$argmin_{(U,V)\in C}\{D_W\left(X,f\left(UV^T\right)\right)+R(U,V)\}$$

其中的损失函数 $D(\cdot,\cdot)$ 一般对于第二个自变量是凸函数，且往往可以分解为矩阵中各个元素上的损失之和。例如，对于常见的加权奇异值分解（Weighted Singular Value Decomposition, WSVD）算法而言，其损失函数为：

$$D_W\left(X,f\left(UV^T\right)\right)=\|W\odot(X-UV^T)\|_{Fro}^2$$

其中，符号⊙表示将两个同维度的矩阵对应元素相乘得到一个新的矩阵，$\|\cdot\|_{Fro}^2$ 则表示矩阵 Frobenius 范数的平方，即矩阵中各个元素的平方和。

选取不同的预测函数 f、权重矩阵 W、损失函数 D_W 和正则化项 R 等各部分，就可以获得很多不同的矩阵分解算法，用以解决不同背景下的个性化推荐问题。

3. 常用矩阵分解算法

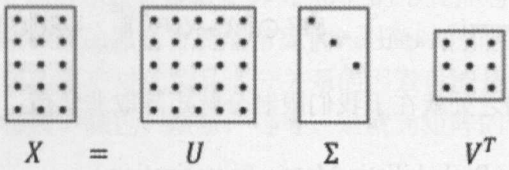
在如上的矩阵分解形式化描述下，我们介绍几种常见的矩阵分解方法，从而帮助读者对矩阵分解在个性化推荐、尤其是矩阵的打分预测任务上的应用有更为深入的了解。

- 矩阵的奇异值分解（Singular Value Decomposition, SVD）

奇异值分解（Wall,2003; Koren,2009）在矩阵计算中具有理论上重要的基础性意义。最原始的矩阵奇异值分解方法具有严格的数学定义，设 $X\in R^{m\times n}$ 是一任意矩阵，矩阵 $U\in R^{m\times r}$ 中的列向量是矩阵 $XX^T\in R^{m\times m}$ 的单位正交特征向量，矩阵 $V\in R^{n\times r}$ 中的列向量是矩阵 $X^TX\in R^{n\times n}$ 的单位正交特征向量，对角矩阵 $\Sigma\in R^{r\times r}$ 中的每一个对角元素 $\sqrt{\sigma}$ 则是与矩阵 U（同时也是与矩阵 V）中的每一个列向量对应的特征值 σ 的平方根，并以从大到小的顺序排列，则原矩阵 X 可表示为 $X=U\Sigma V^T$，其中 Σ 被称为奇异值矩阵。如果我们只保留奇异值矩阵 Σ 中的前 k 个最大的奇异值，同时只保留 U 和 V 中的前 k 个对应的列向量，则新的矩阵 $\hat{X}=U_k\Sigma_kV_k^T$ 即为对原矩阵 X 的一个近似。可以证明，对原矩阵 X 所有秩为 k 的近似中，采用如上 SVD 得到的近似结果可以取得最小的平方误差，即：

$$\hat{X}=U_k\Sigma_kV_k^T=argmin_{rank(\hat{X})=k}\|X-\hat{X}\|_{Fro}^2$$

当然，通过这种方式取得的近似矩阵也就具有最小的 RMSE 值。



然而在实际推荐系统中，我们所要处理的往往是非常稀疏的矩阵，即矩阵中存在大量的未知打分（以 0 值表示）。需要指出的是，这些 0 值并不意味着用户对相应的物品打了 0 分，而仅仅表示用户没有进行相关的打分、或者我们没有观测到相应的打分，因此在计算预测精度时，将这些元素上的预测值也考虑在内并以 0 作为真实值进行评测是不合理的。

基于这一事实，实际推荐系统中所使用的 SVD 算法并不是原始的“精确的”SVD，而是只考虑已观测数据进行模型训练和预测的 SVD 算法，并采用优化的方法求得近似矩阵以应对大规模的稀疏矩阵。考虑如上的 SVD 分解，我们可以用如下的方式将其转化为两个矩阵相乘的形式：

$$\hat{X} = U_k \Sigma_k V_k^T = U_k \sqrt{\Sigma_k} \sqrt{\Sigma_k} V_k^T = \left(U_k \sqrt{\Sigma_k}\right)\left(V_k \sqrt{\Sigma_k}\right)^T = U_k' V_k'^T$$

因此，我们采用低秩近似的方式，用两个秩较低的矩阵的乘积来近似一个秩较高的大规模矩阵，并考虑在已观测的点上对矩阵进行优化，得到如下的 SVD 算法：

$$(U_k, V_k) = argmin_{U \in R^{m \times k}, V \in R^{m \times k}} \{ \left\| W \odot \left(X - UV^T\right) \right\|_{Fro}^2 + \lambda(\|U\|_{Fro}^2 + \|V\|_{Fro}^2) \}$$

其中权重矩阵 W 的元素中相应于原矩阵 X 上的已观测点的取值为 1，而相应于未观测点的元素取值为 0，直观上也就是只考虑原矩阵中已观测点上的预测损失，而正则化项 $\lambda(\|U\|_{Fro}^2 + \|V\|_{Fro}^2)$ 用于最小化模型复杂度，从而减小模型过拟合带来的影响。

- 非负矩阵分解（Non-negative Matrix Factorization, NMF ）

在如上的 SVD 算法中，我们并没有对分解矩阵 $\tilde{U}_k$ 和 $\tilde{V}_k$ 附加其他限制条件，而只是简单地要求其列维度为 k。然而在很多实际应用场景中，我们希望矩阵分解所得到的分解向量满足一定的条件，最常见的就是要求分解矩阵的各个列向量由非负值组成（Lee & Seung, 2001,1999），这是因为在很多场景中（如图像处理、社交网络关系处理、文本处理、概率估计等等）我们所处理的数据均为非负值，采用非负的向量更符合问题的假设，因而往往能取得更好的效果。因此，非负矩阵分解相对于 SVD 算法在包括推荐系统在内的很多实际

系统中获得了更为广泛的应用。典型的非负矩阵分解算法，其优化表达式为：

$$(U_k,V_k)=argmin_{U\in R_+^{m\times k},V\in R_+^{n\times k}}\{\left\|W\odot\left(X-UV^T\right)\right\|_{Fro}^2+\lambda(\|U\|_{Fro}^2+\|V\|_{Fro}^2)\}$$

其中与SVD的不同之处就在于我们限制分解矩阵取非负值：$U\in R_+^{m\times k},V\in R_+^{n\times k}$。

- 概率矩阵分解（Probabilistic Matrix Factorization）

概率矩阵分解（Mnih & Salakhutdinov, 2007,2008）则为矩阵分解算法提供了概率框架下的解释，并试图利用概率模型对原始矩阵中的已观测点进行最大似然估计：

$$p\left(X|U,V,\sigma^2\right)=\prod_{i=1}^{m}\prod_{j=1}^{n}[\mathcal{N}(X_{ij}\,|\,U_i^TV_j,\sigma^2)]^{W_{ij}}$$

其中$\mathcal{N}(x\,|\,u,\sigma^2)$为高斯分布，$W_{ij}$仍然描述原始打分矩阵的数值分布情况，即对应于原始矩阵中的非零值W_{ij}=1，否则W_{ij}=0。同样，我们也可以用概率分布来描述分解矩阵U和V：

$$p\left(U|\sigma_U^2\right)=\prod_{i=1}^{m}[\mathcal{N}(U_i\,|\,0,\sigma_U^2I)],p\left(V|\sigma_V^2\right)=\prod_{i=1}^{n}[\mathcal{N}(V_j\,|\,0,\sigma_V^2I)]$$

这样，我们在已观测数据上最大化如下的对数似然概率：

$$lnp\left(U,V|X,\sigma^2,\sigma_U^2,\sigma_V^2\right)=-\frac{1}{2\sigma^2}\sum_{i=1}^{m}\sum_{j=1}^{n}W_{ij}\left(X_{ij}-U_i^TV_j\right)^2-\frac{1}{2\sigma_U^2}\sum_{i=1}^{m}U_i^TU_i-\frac{1}{2\sigma_V^2}\sum_{j=1}^{n}V_j^TV_j+C$$

容易看出，该形式同样可以表达为统一的矩阵分解形式，如下面的最小化问题所示：

$$(U,V)=argmin_{U\in R^{m\times k},V\in R^{n\times k}}\left\{\frac{1}{2}\left\|W\odot\left(X-UV^T\right)\right\|_{Fro}^2+\frac{\lambda_U}{2}\|U\|_{Fro}^2+\frac{\lambda_V}{2}\|V\|_{Fro}^2\right\}$$

- 最大间隔矩阵分解（Maximum Margin Matrix Factorization）

如上所介绍的常用矩阵分解算法所隐含的基本假设均为“低秩近似”假设：虽然一个大规模的稀疏矩阵其原始形式可能具有较高的秩，但我们认为可以用一个秩较低的矩阵来对原始矩阵进行近似和重构。

在如上的矩阵分解中，原始矩阵的行数和列数（m 和 n）可以高达上千万甚至上亿的量级，但是我们所采用的分解矩阵U和V的列数被限制在k维，而k在实际系统中不过是几十至几百的量级，而近似矩阵UV^T的秩一定小于或等于 k，相对于原始矩阵而言大大降低。其中隐含的直观意义在于，虽然原始矩阵的规模和维度非常庞大，但我们认为用来描

述这些数据的规律、结构和参数的数量是有限的，且最多用可 k 个维度就可以较为完备地刻画出来，这 k 个维度的参数就用超线性参数矩阵 U 和 V 的形式描述出来。例如一个化妆品购物网站所包含的用户数量和商品数量可能非常庞大，因而对应的用户-物品稀疏矩阵也具有庞大的规模。但是描述网站用户对化妆品评分关系的因素可能是为数不多的有限个，例如化妆品的品牌、价格、包装、颜色、质量，等等。这就为矩阵的低秩分解和近似提供了直观基础。

但是基于低秩近似的矩阵分解算法存在的一个重要问题就是，在实际操作中往往难以确定选择多少个维度来进行分解，因为在一个实际应用中，设计人员往往很难估计决定系统规律的参数维度到底有多少个。使用的维度过少则可能无法完全描述用户的偏好信息，从而造成预测精度的下降；而使用的维度过多则会带来巨大的计算负载、甚至带来过拟合。

最大间隔矩阵分解（Srebro,2004; Rennie,2005）则突破传统的低秩近似假设，而采用低范数近似的理念对原始矩阵进行分解和预测，从而绕过选取特定维度进行分解的必要。为了简化问题描述和符号表示，我们在这里采用两类标签的矩阵。假设原始矩阵 $Y \in \{\pm 1\}^{m \times n}$，对 Y 的近似矩阵为 X，则最大间隔矩阵分解最小化如下的优化目标：

$$minimize \sum_{i,j} W_{ij} \cdot \max(0, 1 - X_{ij} Y_{ij}) + \lambda \| X \|_{\Sigma}$$

其中的正则化项 $\| X \|_{\Sigma}$ 为矩阵的核范数(Nuclear Norm)，它表示一个矩阵的各个特征值之和，用来描述和控制模型的复杂度。可以证明，矩阵的核范数可以等价表示为如下的形式：

$$\| X \|_{\Sigma} = min_{X=UV^T} \|U\|_{Fro} \|V\|_{Fro} = \min_{X=UV^T} \frac{1}{2} (\|U\|_{Fro}^2 + \|V\|_{Fro}^2)$$

即一个矩阵的核范数等于其所有可能的分解矩阵（不对分解的维度进行具体限制）的 Frobenius 范数中的最小值。正是基于这一定理，我们可以绕过显式的矩阵分解，转而采用直接最小化核范数的方式使得在全部可能的维度下寻找最优解成为可能。

6.3.4 推荐的可解释性

除了给出推荐列表之外（如图 6.10 所示），推荐理由的构建也是推荐系统的重要组成部分和研究方向。相关研究指出：在推荐系统中提供直观合理的推荐理由可以大大提高用户对推荐结果的接受度，同时也有助于在很多其他方面增强用户体验，如系统的透明性、

可信性、有效性、推荐效率、用户满意度，等等（Herlocker, Konstan and Riedl, 2000; Tintarev and Masthoff, 2007）。但是推荐理由的构建往往需要与系统所使用的推荐算法相匹配，并且往往依赖于所使用的推荐算法。一般而言，常用的基于隐变量的个性化推荐算法（如上所述的常用矩阵分解算法），由于本质上变量意义的隐含性，难以为推荐结果给出直观易懂的解释，这也是基于隐变量的个性化推荐方法缺点之一。

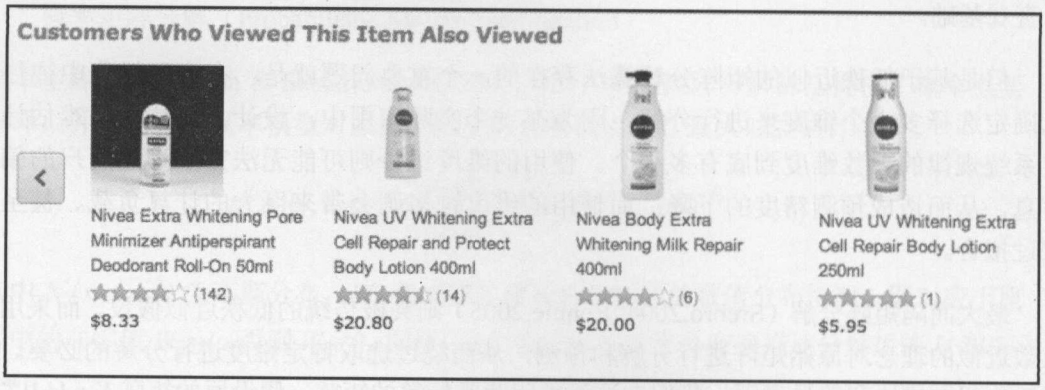


图 6.10 推荐列表

个性化推荐算法中推荐理由的构建大致可以分为两种类型，一种是构造于模型之后的解释（Post-model Explanation），另一种是由模型内生的解释（Intro-model Explanation）。

模型后解释的方法在构建推荐列表的过程中先不考虑推荐理由，而是在推荐列表构建完成之后为算法给出的推荐“寻找”一个看上去合适的推荐理由，而该推荐理由与给出该推荐的具体算法可以没有必然联系，甚至完全无关。例如，我们首先可以通过非负矩阵分解对用户-物品评分矩阵进行打分预测，然后对目标用户选取预测分值最高的前几个未浏览商品构建推荐列表。当我们需要对某一个被推荐出来的物品进行解释时，可以在系统所收集的大规模用户行为信息中统计浏览了该物品的用户数量，然后告诉目标用户“有百分之几的用户浏览了该商品”作为推荐理由，这也是很多实际系统（尤其是网络购物系统）中常见的推荐理由之一。可见，在该过程中最终我们给用户展示的推荐理由与生成该推荐的实际算法没有必然联系，但是该推荐理由来自系统收集统计的大规模真实用户行为信息，因此又是完全真实和可接受的。这样的推荐理由简单、直观、容易构造，因此在实际系统中得到了广泛的应用。

然而模型后解释的方法毕竟脱离了推荐列表的真实构建过程，因此系统给出的推荐理由可能与推荐结果脱节、难以非常精确地描述为何该物品被推荐给了用户。实际上，如果推荐理由的构建可以与所使用的推荐算法相辅相成，就可以在推荐算法执行的过程中收集有效的信息，跟踪一个物品被算法推荐给特定用户的具体机制和过程，从而向用户展示更为具体、细致、有说服力的推荐理由，模型内生的推荐理由构建则致力于给出这样高可信

度的推荐理由。一个简单地例子是前面所介绍的基于物品的推荐算法。在基于物品的推荐中，我们对每一个用户计算其未浏览过的物品与已浏览（购买）物品的加权相似度，并给出相似度最高的几个未浏览物品作为推荐结果，相应的推荐理由则为“被推荐的物品与您曾经购买过的某（些）物品相似”，我们还可以具体给出这些相似的物品，从而进一步增强推荐理由的可信度。这样的推荐理由就是由推荐模型本身派生的，与所使用的具体推荐算法紧密相关。从某种意义上讲，使用什么样的推荐算法，就决定了被推荐物品的推荐理由如何。更为复杂一些，（Zhang, et al. 2014）中与隐变量分解模型相对应地提出了显式变量分解模型，从用户的评论文本中抽取出诸如“价格”、“质量”、“颜色”等特定领域商品的属性词，并将其作为显式的变量、与隐变量一起加入到非负矩阵分解框架中，从而不仅可以预测用户在不同物品上的打分，还可以根据用户在不同属性词上对应权重的不同来确定到底是哪些属性决定了用户的最终打分，从而给出诸如“该推荐是因您比较关心某属性、而该商品在该属性上表现较好”这样具体明确的推荐理由，从而让推荐结果更为真实可靠，吸引用户点击甚至采纳系统给出的推荐。

模型后解释和模型内生的解释各有优点、也各有自己的缺点和局限性。模型后解释由于不依赖于具体所使用的模型，因而推荐理由的构建更为灵活多样、可选择性多，当推荐算法给出了推荐结果之后，我们可以进一步充分利用系统中包含的用户、物品信息和用户浏览、点击历史记录，根据不同的目的设计多种不同的推荐理由；其缺点则是推荐理由与实际情况不一定相符合，因为推荐理由与产生该推荐结果的算法未必有任何联系。模型内生的推荐则考虑了产生推荐结果的具体算法，通过分析算法的执行过程和给出推荐结果的内在逻辑构建与之相匹配的推荐理由，因此推荐理由往往更为具体、细致、有说服力，然而正是由于推荐理由受限于具体所使用的推荐算法，也造成推荐理由模式比较单一。

6.3.5 推荐算法的评价

一个推荐算法的好坏必须用可靠的评价指标去度量，从而帮助我们了解和改进系统的性能。评价指标主要包括“线下评价指标”和“线上评价指标”。线下评价指标包括诸如根均方差（Root Mean Square Error, RMSE）、平均绝对误差（Mean Absolute Error, MAE）、归一化折扣增益值（Normalized Discounted Cumulative Gain, NDCG）、平均准确率（Mean Average Precision, MAP）、准确率（Precision）、召回率（Recall）、F_1 值（F_1-measure）等；线上评价指标又包括成交转化率、用户点击率，等等。在这里，我们主要对常用的线下评价指标进行总结概括（刘建国、周涛等，2009）。

一个比较有意思的事情是，在线视频提供商 Hulu 在（Zheng, et al. 2010）中讨论了点击率是否适用于评测推荐系统，报告认为在搜索领域被广泛认可或验证了的位置偏置

（Position Bias）假设（即排在靠前位置的搜索结果得到的点击会比靠后位置的结果多得多）并不适用于推荐系统，他们的实验表明推荐产品的排列位置对点击影响甚微，因此在以NDCG为指标的离线测评中性能好的算法，在在线测评中点击率有可能反而比较低。

目前评估推荐系统的线下指标大致可分为“准确性（Accuracy）”与“可用性（Usefulness）”两个方面。其中准确性衡量的是推荐系统的预测结果与用户行为之间的误差，还可以再细分为“预测准确度（Prediction Accuracy）”和“决策支持准确度（Decision-Support Accuracy）”。预测准确度又可分为“评分预测准确度”、“使用预测准确度”和“排序准确度”等，以MAE、RMSE等为常用的统计指标，来计算推荐系统对消费者喜好的预测与消费者实际的喜好间的误差平均值；而决策支持准确度则以关联度（Correlation，包括Pearson、Spearman、Kendall Tau等相关系数）、准确度（Precision）、召回率（Recall）、F1值（F1-measure）、ROC曲线（Receiver Operating Characteristic）、曲线下面积（Area Under Curve, AUC）等为主要工具。

1. 评分预测的评价

根均方差RMSE是最流行的度量指标，它描述了算法预测的打分与用户的真实打分之间的差距。优化RMSE度量指标，实际上就是要预测用户对每个商品的评分。

$$RMSE = \sqrt{\frac{\sum_{r_{ij} \in \hat{S}} (r_{ij} - \hat{r}_{ij})^2}{|\hat{S}|}}$$

其中 r_{ij} 是用户 i 对物品 j 的真实打分，$\hat{r}_{ij}$ 为算法给出的预测打分，$|\hat{S}|$ 表示测试数据集所包含测试样例的个数。例如2007~2009年间著名的Netflix Prize竞赛，就是以RMSE为评价指标，竞赛者比Netflix公司使用的推荐系统算法CineMatch预测误差低百分之十，就可获得百万美元大奖，如图6.11所示。

图6.11 比Netflix公司使用的推荐系统算法CineMatch误差低10%的竞赛者获得了百万美元大奖

另一个常用的度量平均绝对误差 MAE 则直接计算预测值与真实值之间的误差绝对值：

$$MAE = \frac{\sum_{r_{ij} \in \hat{S}} | r_{ij} - \hat{r}_{ij} |}{| \hat{S} |}$$

RMSE 和 MAE 两个指标虽然类似，但是两者相比前者对大误差更为敏感，对预测算法的评价也更为严格。

RMSE 和 MAE 仅度量误差幅度，容易理解且计算方式也不复杂，但其缺点也正是失之于简单，过度简化事实，在有些场合可能不能说明问题。假设用户的真实打分为 3，那么预测打分为 1 或 5 的差别都是 1，但实际意义却截然相反。在这种情况下可以定义适当的扭曲程度度量（Distortion Measure）来代替差值以改进度量方法。

2. 推荐列表的评价

除打分预测之外，推荐系统最终要给用户提供一个个性化的推荐列表，对该推荐列表的效果评价是评测推荐算法实际效果的重要部分。

与信息检索理论同源的准确率(Precision)、召回率(Recall)、F1 值(F1-Measure)评价指标是评价推荐列表最基本也是最常用的指标。假设对于一个用户而言，用作测试样本的购买记录集合为 S_{test}，而推荐系统为用户构造的推荐列表集合为 S_{rec}，则准确率、召回率和 F 值的计算如下：

$$P = \frac{| S_{test} \cap S_{rec} |}{| S_{rec} |}, R = \frac{| S_{test} \cap S_{rec} |}{| S_{test} |}, F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

对于推荐列表的长度确定为 n 的场合，我们可使用 $Precision@n$(P@n)、$Recall@n$(R@n)和 $F_1@n$ 来度量，而对于推荐数量未指定的场合我们则可以使用 PR 曲线(Precision-Recall 曲线)或 ROC(Receiver Operating Characteristic)曲线来描述推荐出正确物品的比例，其中 PR 曲线强调被推荐的物品有多少是正确的，而 ROC 曲线则强调有多少用户不喜欢的物品却被推荐了出来。与其相对应的 AUC(Area Under the ROC Curve)指标则对不同 Precision-Recall 取值下的推荐效果给出一个综合的评价，AUC 越大表示系统能够推荐出越多正确的物品。已经有研究人员验证，大部分状况下计算 ROC 和 Precision-Recall 时，会得到相同的混淆矩阵(Confusion Matrix)，而且从其中一个曲线可以推演出另外一种曲线的状况。不过 PR 曲线比较适合于数据分布高度不平均（highly-skewed）的情况，因此在实际应用中要根据推荐系统选择相应的评估方式。

以上的评价指标实际上只考虑了推荐集合的正确与否，而没有关注被推荐物品的排序。

实际上，同样的推荐物品集合，我们希望正确的物品越靠前越好。因此，一个更为合理的评价方式是把被推荐物品的位置信息也考虑在内，平均准确率 MAP 即评估用户期望的相关结果是否尽可能排在前面。MAP 是信息检索中解决 PRF 指标的不足而提出的，单个主题的平均准确率是每篇相关文档检索出后的准确率的平均值，主集合的平均准确率（MAP）是每个主题的平均准确率的平均值。MAP 是反映系统在全部相关文档上性能的单值指标。系统检索出来的相关文档越靠前，则 MAP 就可能越高。对于 N 个推荐列表，假设每一个列表的长度均为 n，则 MAP 可以做如下表示：

$$MAP=\frac{1}{N}\sum_{i=1}^{N}\frac{\sum_{j=1}^{n}P_i(k)\cdot\delta_{ij}}{\sum_{j=1}^{n}\delta_{ij}}$$

其中 $P_i(k)$ 表示第 i 个推荐列表在位置 k 的准确率，δ_{ij} 为一个示性函数，表示列表 i 中的第 j 个项是否为正确的推荐，当它是一个正确的推荐时 $\delta_{ij}=1$，否则 $\delta_{ij}=0$。

另一个同样考虑未知因素且经常被用来评价推荐列表质量的指标为归一化折扣增益值 (Normalized Discounted Cumulative Gain, NDCG)，同样考虑 N 个推荐列表，且每一个列表的长度均为 n，示性函数为 δ_{ij}，则 NDCG 可以通过如下的方式计算：

$$NDCG=\frac{1}{N}\sum_{i=1}^{N}\frac{1}{IDCG_i}\sum_{j=1}^{n}\frac{2^{\delta_{ij}}-1}{\log_2(j+1)}$$

其中的 $IDCG_i$ 是第 i 个推荐列表所有可能取到的最大的 DCG 值，即当该列表中所有正确的物品均排在列表最前面时 $\sum_{j=1}^{n}\frac{2^{\delta_{ij}}-1}{\log_2(j+1)}$ 这部分的值，其作用是保证 NDCG 的理想值为 1，从而便于比较。

在科学研究和实际系统中，我们必须按照任务的实际需要来选择合适的、有说服力和代表性的指标来进行算法验证和系统评价，一般情况下，可以选择 RMSE 或 MAE 指标来验证算法在打分预测任务上的表现，用 Precision、Recall、F-measure、MAP 和 NDCG 指标来验证算法在推荐列表构建任务上的表现；为了验证算法的线上效果，还可以进一步采用点击率等线上指标来评价在实际系统中用户点击推荐结果的真实情况。

6.3.6 我们走了多远

虽然经历了几十年的研究和发展，推荐系统已经成为各种现代网络应用中不可或缺的

组成部分，但是推荐系统的研究和应用仍然面临着很多重要而急迫的挑战，推荐系统的应用形式和场景也蕴含着更多的可能。在本节，我们总结归纳目前推荐系统在研究和应用方面所面临的一些重要问题，同时指出推荐系统在未来研究和应用的一些潜在方向，以使读者对推荐系统的未来发展拥有一些认识。

1. 推荐系统面临的问题

冷启动（Cold-Start）问题是困扰研究界和产业界多年的重要问题。对于一个全新的网络用户，由于系统中尚没有任何关于该用户的商品购买或浏览交互等信息可以用来分析其个性化偏好和需求，因此无法向其提供个性化的推荐列表。该问题在传统的基于数值化评分的个性化推荐方法中尤为突出，并与数据的稀疏性问题互为因果，这是由于网站内的新注册用户往往只对非常少量的数个商品给出过数值化的评分，因此很难通过如此少数的评分分析用户的偏好和需求。另外，在大数据环境下，数据的稀疏性显得愈加明显和严重，这进一步加重了冷启动问题对实际系统带来的负面影响。

目前，解决冷启动问题的方法主要包括：降维技术（Dimensionality Reduction），通过 PCA、SVD 等技术来降低稀疏矩阵的维度从而为原始矩阵求得最好的低维近似，但是实际系统中庞大的数据规模使得降维过程存在大量运算成本，并有可能影响预测和推荐效果；使用混合推荐模型的方法，通过取长补短弥补其中某种方法的问题；加入用户画像信息和物品属性信息，例如通过使用用户资料信息来计算用户相似度，或者使用物品的内容信息来计算物品相似度，进一步与基于打分的协同过滤方法相结合以提供更为准确的推荐。

另外，推荐系统中的小众用户（Gray Sheep）问题是限制系统在小众用户上取得较好性能的重要问题，该问题主要表现为有些人的偏好与任何人或绝大多数人都不同，因而难以在大规模数据上采用协同过滤的方式为该用户给出合理的推荐。目前，小众用户推荐一般采用混合式的推荐模型来解决，例如最常见的结合内容的和协同过滤的混合式推荐方式，挖掘小众用户在感兴趣的物品上的内容信息，并进一步结合可能取得的相似用户行为信息给出推荐。然而，该方案在解决小众用户推荐的问题上还远远不够，由于长尾效应的存在，系统在小众用户上的性能对整体能取得的性能有较大的影响，因此小众用户推荐的问题需要进一步的研究和实践。

个性化推荐的可解释性长期以来是困扰学术界和实际应用的重要问题，随着推荐算法变得越来越复杂和隐性变量方法的大量使用，算法所给出的推荐列表往往并不能得到较为直观的解释，也就难以让用户理解为什么系统会给出该物品作为推荐而不是其他物品。当前的实际系统中往往简单地给出“看过该物品的用户也看过这些物品”作为推荐

理由，然而这样的推荐理由往往无法令人信服，从而降低了用户点击和接受推荐结果的潜在可能性。在跨领域的异质推荐背景下，推荐结果的可解释性显得更为重要，因为缺乏直观可信的推荐理由将难以说服用户进入新的甚至陌生的网站查看异质推荐结果。如何将推荐理由的构建与系统所使用的推荐算法紧密结合，从而得到更为细致、准确、有说服力的推荐理由，进而引导用户查看甚至接受系统给出的推荐，是学术研究和实际系统都需要考虑的重要问题。

推荐系统如何应对恶意攻击（Shilling Attack）也是实际系统中需要解决的重要问题，该问题实际上是推荐系统中的反垃圾（Anti-Spam）问题。例如，有些用户或商家会频繁为自己的物品或者对自己有利的物品打高分，而恶意为竞争对手的物品打低分，甚至注册大量的系统账号来干预某物品的得分，从而达到人工干预推荐系统推荐效果的目的，这会影响协同过滤算法的正常工作。该问题的被动的解决办法可以是采用基于物品的（Item-based）推荐，因为在恶意攻击的问题上，基于物品的推荐往往能比基于用户的推荐具有更好的鲁棒性，因为作弊者总归是较少数，在计算物品相似度的时候影响较小，然而在基于用户的推荐中很有可能为正常用户计算得到的近邻用户都是作弊用户，从而使正常用户所得到的推荐列表受到干扰。当然，我们也可以采用主动的解决办法，设计有效的垃圾用户识别技术来识别和去除作弊者的影响。

除此之外，推荐系统的研究和应用中还面临很多其他的问题和挑战，如隐私问题、噪声问题、推荐的新颖性，等等。在这些问题上的研究急需进一步投入更多的研究和实践，从而不断完善推荐系统的性能和应用场景。

2. 推荐系统的新方向

长期以来，推荐系统的各种算法和研究都是基于（Resnick & Iacovou, 1994）所提出的数值化打分矩阵的形式化模型，该模型的核心是以用户打分为基础，而少有对基于用户文本评论语料进行个性化推荐的研究。基于文本评论的个性化推荐被很多论文提到，但是研究并不深入，这一方面限于文本挖掘技术和研究遇到很多难点，另一方面限于之前网络上所积累的文本信息还不够多。然而伴随着 Web 2.0 网络的兴起，互联网上所积累的用户文本信息越来越多，已经成为一种不可忽略的信息来源，如电子购物网站中的用户评论、社交网络中的用户状态，等等。这些文本信息对于了解用户兴趣、发掘用户需求有极其重要的作用，如何充分利用这些数值评分之外的文本信息进行用户建模和个性化推荐具有重要的意义。

推荐系统与用户的交互方式也是相关领域内研究的热点方向。目前常见的实际系统一般以推荐列表的形式给出推荐，然而一些研究表明，即便是同样的打分和评价系统，如果

展示给用户的方式不同，也会对用户的使用、评价、效果产生一定的影响。比如 MovieLens 小组（Cosley, et al. 2003）第一次研究了用户打分区间、连续打分还是离散（如星标）打分、推荐系统主动欺骗等对用户使用推荐系统造成的影响。与搜索引擎一样，推荐系统的界面设计和交互方式也越来越受到研究人员的关注。

长尾效应在推荐系统中的理解和应用可以为进一步提高系统的推荐效果打开新的窗户。一个实际推荐系统的性能不能直接以预测评分的精确度来测量，而应该以用户的满意度来考虑。推荐系统应该以“发现”为核心终极目标，而现存的一些推荐技术通常会倾向于推荐流行度很高的，用户已经知道的物品。这样存在于长尾中的物品也就不能很好地推荐给相应的用户。但是，这些长尾物品通常更能体现用户的兴趣偏好。所以，在推荐系统的设计过程中，不仅要考虑预测的精度，而且还要考虑用户真正的兴趣点在哪里。最近，研究人员已开始考虑长尾效应在推荐系统设计过程中的应用，并考虑如何将长尾物品推荐给用户，以及如何为小众用户推荐合适的物品。

另外，在如今的网络化的大潮中，我们需要前瞻性地看到一个重要而急迫的问题，即在越来越多的生活项目日益网络化的同时，也在网络上造成了一个个的信息孤岛：每一个网络应用平台拥有用户在该平台或该领域内的行为信息、了解用户在该平台和领域内的行为偏好，从而可以在该领域内给出个性化的专业服务；然而在不同平台和领域之间、尤其是异质领域（如视频和购物）之间，用户的行为线索并没有被打通，每一个平台和领域也没有其他平台和领域的用户行为信息，也就难以给出平台之外、其他领域的个性化服务。这些独立的信息孤岛将网络用户原本完整而流畅的生活时间线割裂开来，未能形成浑然一体的个性化服务流程贯穿网络用户生活的始终，使得互联网本应在人们日常生活中所起的重要甚至核心的作用大打折扣。

因此，如何使互联网所连接的各个系统能够协作式地发掘用户潜在需求、适时地给出跨领域的异质推荐结果和个性化服务成为推荐系统向通用推荐引擎方向发展的重要问题和研究前沿，并将极大地降低人们使用互联网的时间和精力成本，免去在各个独立服务之间进行切换和查找的麻烦。更重要的是，不同类型的异质商品或服务之间的信息联通和相互推荐，这其中就蕴含着全新的互联网运营和盈利模式。例如，通过从历史数据中进行任务挖掘，旅行机票订购网站可以通过异质推荐为酒店预订、车辆租赁、团队预订等多种潜在的关联网站带来流量，并从中获得额外收益；视频服务商甚至可以通过异质推荐给出来自购物网站的商品推荐，从而实现虚拟产业收入与实物商品收入的结合，这对促进产业协作发展和产业整合具有重要意义。

从这一角度来看，推荐系统以及推荐系统背后所隐涵的用户个性化建模的思想，将在不久的未来以平台化应用的形式全面渗透到桌面端、手机端、甚至操作系统中，典型的例

子有个人手机助手、个性化办公助手等平台化个性化工具的蓬勃发展。在不久的将来，手机电脑等基于互联网的终端将回更加智能、更加懂你，为用户提供个性化的办公和娱乐体验；同时以物联网、智能家居为典型代表的线下终端，也将更加懂你，在真实物理世界的日常生活中时时刻刻为我们提供准确贴心及时的个性化生活服务。一言以蔽之，个性化推荐技术及其试图去理解用户的基本思想将作为人工智能的核心环节，在未来智能生活的大潮中发挥重要作用。

6.4 参考文献

[1] （Adomavicius 2005）G. Adomavicius, “Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions”, IEEE Transactions on Knowledge and Data Engineering, 2005.

[2] （Resnick, et al. 1994）P. Resnick, N. Iacovou, etc. “GroupLens: An Open Architecture for Collaborative Filtering of Netnews”, Proceedings of ACM Conference on Computer Supported Cooperative Work, CSCW 1994. pp.175-186.

[3] （Sarwar, et al. 2001）B. Sarwar, G. Karypis, J. Konstan, J. Riedl, “Item-Based Collaborative Filtering Recommendation Algorithms”, Proceeding of the 10th international conference on World Wide Web, WWW 2001.

[4] （Sugiyama, et al. 2004）K. Sugiyama, K. Hatano, M. Yoshikawa, Adaptive Web Search based on User Profile Constructed without Any Effort from Users. Proceeding of the 13th international conference on World Wide Web, WWW 2004.

[5] （Wietsma & Ricci 2005）R. Wietsma, F. Ricci, “Product Review in Mobile Decision Aid Systems”, Proceeding of Workshop Pervasive Mobile Interaction Devices. Pervasive 2005.

[6] （Ricci & Wietsma 2006）F. Ricci, R. Wietsma, “Product Reviews in Travel Decision Making”, Information and Communication Technologies in Tourism: Proc. Int’I Conf.

[7] （Aciar, et al. 2007）S. Aciar, D. Zhang, S. Simoff, J. Debeham. Informed Recommender: Basing Recommendations on Consumer Product Reviews. Recommender Systems. 2007.

[8] （Ziegler, et al. 2005）C Ziegler, S. M. McNee, etc. Improving Recommendation List Through Topic Diversification. Proceeding of the 15th international conference on World Wide Web. WWW 2005.

[9] （Tintarev & Masthoff 2000）N. Tintarev, J. Masthoff, A Survey of Explanations in Recommender Systems, Proceedings of the 2000 ACM conference on Computer supported cooperative work, CSCW 2000.

[10] （Pazzani 1999）M. J. Pazzani, “A framework for Collaborative, Content-Based and Demographic Filtering”, Artificial Intelligence Review, Springer, 1999

[11] （Gunawardana & Shani 2009）A. Gunawardana, G. Shani, A Survey of Accuracy Evaluation Metrics of Recommendation Tasks. Journal of Machine Learning, Volume 10, pp.2935-2962

[12] （Herlocker, et al. 1999）J. L. Herlocker, J. A. Konstan, Al Borchers, J. Riedl, An algorithmic framework for performing collaborative filtering, Proceedings of the 22nd international SIGIR conference on Research and development in information retrieval, SIGIR 1999.

[13] （Kautz, et al. 1997）H. Kautz, B. Selman, M. Shah, Referral Web: combining social networks and collaborative filtering, Communications of the ACM, Volume 40 Issue 2, March 1997

[14] （Sarwar, et al. 2000A）B. M. Sarwar, G. Karypis, etc, Application of Dimensionality Reduction in Recommender System – A Case Study, DTIC Document, 2000

[15] （George & Merugu 2005）T. George, S. Merugu, A Scalable Collaborative Filtering Framework based on Co-clustering, Proceedings of the Fifth IEEE International Conference on Data Mining, ICDM 2005

[16] （Melville, et al. 2002）P. Melville, R. J. Mooney, R. Nagarajan, Content-Boosted Collaborative Filtering for Improved Recommendations, Proceedings of American Association for Artificial Intelligence, AAAI 2002

[17] （Sarwar, et al. 2000B）B. M. Sarwar, G. Karypis, J. Konstan, J. Riedl, Analysis of Recommendation Algorithms for E-Commerce, Proceedings of the 2nd ACM conference on Electronic commerce, ACM EC 2000

[18] （O'Cornor, et al., 1999）M. O’Cornor, Jon Herlocker, Clustering Items for Collaborative Filtering, Proceedings of the ACM SIGIR Workshop, SIGIR 1999

[19] （Zhou, et al. 2011）K. Zhou, S. Yang, H. Zha, Functional Matrix Factorizations for Cold-Start Recommendation, Proceedings of the 34th Annual International ACM SIGIR Conference, SIGIR 2011

[20] （Burke 2002）B. Burke, Hybrid Recommender Systems: Survey and Experiments, User Modeling and User-Adapted Interaction, 12: 331-370, 2002

[21] （刘建国，周涛等 2009）刘建国，周涛等，个性化推荐系统评价方法综述，复杂

系统与复杂性科学第6卷第3期,2009

[22] （Aciar, et al. 2006）S. Aciar, D. Zhang, S. Simoff, J. Debenham， Recommender System Based on Consumer Product Reviews, Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2006

[23] （Aciar, et al. 2007）S. Aciar, D. Zhang, S. Simoff, J. Debenham, Informed Recommender: Basing Recommendations on Consumer Product Reviews, Recommender Systems, June 2007

[24] （Truong, et al. 2007）Truong, K.Q., Ishikawa, F., Honiden, S.Improving Accuracy of Recommender System by Item Clustering, IEICE TRANSACTIONS on Information and Systems, E90-D-I, 2007

[25] （Park, et al. 2008）Y. Park, A. Tuzhilin, The long tail of recommender systems and how to leverage it, RecSys, 2008.

[26] （Ishikawa, et al. 2008）M. Ishikawa, P. Geczy, N. Izumi, T. Yamaguchi, Long Tail Recommender Utilizing Information Diffusion Theory, In WI-IAT, 2008

[27] （Cosley, et al. 2003）D. Cosley, S. K. Lam, I. Albert, J. A. Konstan, J. Riedl, Is Seeing Believing? How Recommender Interfaces Affect Users' Opinions, Proceedings of the 2003 International Conference on Human Factors In Computing Systems, CHI 2003

[28] （Wall, et al. 2003）Wall M E, Rechtsteiner A, Rocha L M. Singular value decomposition and principal component analysis. A practical approach to microarray data analysis. Springer US, 2003: 91-109.

[29] （Koren, et al. 2009）Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. Computer, 2009（8）: 30-37.

[30] （Lee & Seung 2001）Lee D D, Seung H S. Algorithms for non-negative matrix factorization. Advances in neural information processing systems. 2001: 556-562.

[31] （Lee & Seung 1999）Lee D D, Seung H S. Learning the parts of objects by non-negative matrix factorization. Nature, 1999, 401（6755）: 788-791.

[32] （Mnih & Salakhutdinov 2007）Mnih A, Salakhutdinov R. Probabilistic matrix factorization. Advances in neural information processing systems. 2007: 1257-1264.

[33] （Salakhutdinov & Mnih 2008）Salakhutdinov R, Mnih A. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. Proceedings of the 25th international conference on Machine learning. ACM, 2008: 880-887.

[34] （Srebro, et al. 2004）Srebro, Nathan, Jason Rennie, and Tommi S. Jaakkola. Maximum-margin matrix factorization. Advances in neural information processing systems. 2004.

[35] （Rennie & Srebro 2005）Rennie, Jasson DM, and Nathan Srebro. Fast maximum margin matrix factorization for collaborative prediction. Proceedings of the 22nd international conference on Machine learning, 2005.

[36] （Herlocker, et al. 2000）J. Herlocker, J. Konstan, and J. Riedl. Explaining collaborative filtering recommendations. CSCW, 2000.

[37] （Tintarev & Masthoff 2007）N. Tintarev and J. Masthoff. A Survey of Explanations in Recommender Systems. ICDE, 2007.

[38] （Zhang, et al. 2014）Y. Zhang, G. Lai, M. Zhang, Y. Zhang, Y. Liu and S. Ma. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval. ACM, 2014.

[39] （Zheng, et al. 2010）Zheng H, Wang D, Zhang Q,, et al. Do clicks measure recommendation relevancy?: an empirical user study. Proceedings of the fourth ACM conference on Recommender systems. ACM, 2010: 249-252.

第 7 章

情感分析与意见挖掘——计算机如何了解人类情感

喜、怒、哀、惧、爱、恶、欲七者弗学而能。

——《礼记》

7.1 概述

人类情绪、情感和意见是人类的重要特征，如何让计算机了解人类的喜怒哀乐，根据互联网大数据定量分析用户情感，是大数据智能的重要体现。本章将着重介绍情感分析和意见挖掘的相关理念和技术。

在很多上班族中流传着“星期一综合征”这样一种说法，即周末结束，周一开始上班时，刚刚从放松状态回复到紧张的工作中来，心情焦虑、烦躁。还有人据此画了漫画（如图 7.1）：周一、周二心情焦躁；周三、周四逐渐看到了周末来临的希望，一直到周五、周六心情逐渐变好，到周六时的开心程度达到高峰；而到周日由于想到第二天就要上班，心情又有所回落。身边人有这样的感受不假，但是这样的现象普遍吗？这样一个社会学、心理学的现象，在大数据时代，可以方便地利用计算机科学加以论证。看到本章最后，读者朋友将会得到答案。

图 7.1 网友绘制的一周心情变化图

除了网络用户发表的个人情感信息，很多用户产生数据（User-Generated Content，UGC）网站的评论信息也是重要的用户意见集散地，同样吸引了业界和研究领域的注意。

著名的电子商务平台“淘宝网”的评价制度就是一个典型的例子。买家和卖家在交易完成后，可以根据交易情况给对方做出评价：好评、中评、差评。对卖家而言，如果得到的“好评”多，就体现出自己口碑好、有诚信，容易吸引更多的买家。如图 7.2、图 7.3 所示，淘宝网根据这些评价数目设立了“好评率”这一指标，并可以直接看到一家店铺的用户评分总体情况，这就帮助新的买家在消费行为之前做出更好的选择。这些指标如此重要，因此很多商家为了得到用户的好评，不但在售前会有专人和客户进行交流，甚至可能提供附带赠品、包邮费等“小恩小惠”来讨得客户的欢心；如果得到了差评，一些商家会再和客户沟通，希望客户修改评价；有的不法商家甚至可能会对客户软硬兼施，迫使其改为更好的评价。这些现象反映出这一评价体系的重要作用。这种完全由网站用户提供的信息，成为影响用户行为的关键一环。

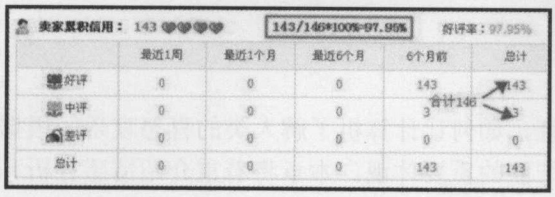


图 7.2 淘宝网评价体系的好评率指标

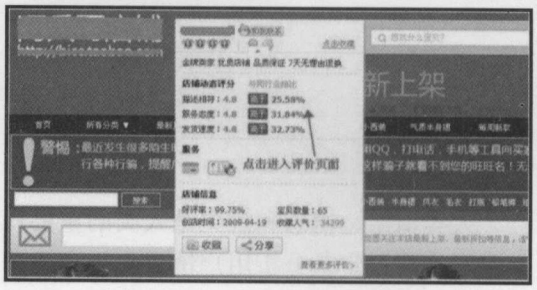


图 7.3 淘宝网店铺评分信息

通过上面的例子，我们看到淘宝网作为平台服务提供方，并不需要直接对买家和卖家的信用做出评价，而是通过广大用户的群体智慧（Wisdom of the Crowd）来形成评价。这是 Web 2.0 时代互联网服务的重要特点。类似地，一些餐馆评价网站、书评影评交流网站，都有用户的评价、打分功能。用户通过阅读他人评价，可以决定自己是否要去某家餐馆吃饭、点哪些招牌菜，决定自己要看哪部电影、读哪本书；通过给出自己的评价，可以让自己的观点影响其他人，也可以找到有相同口味、相同审美的同好。这种现象在大数据时代越来越普遍。据 2014 年 1 月的《中国互联网络发展状况统计报告》（CNNIC 2014），截至 2013 年 12 月，我国网络购物用户规模达到 3.02 亿，使用率为 48.9%。这些用户表达出的观点和潮流，在社会中的地位不可忽视。

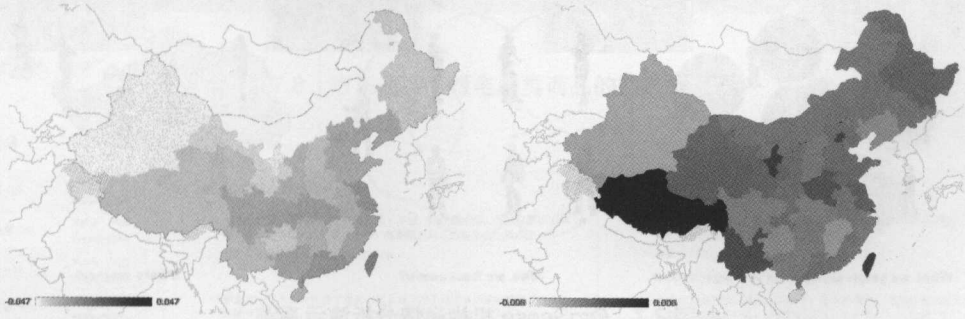
用户表达主观意见的文本，与新闻报道、百科知识文章不同，对社会的影响更直接。因此，在互联网大数据中，对这些主观性文本的定性、定量分析十分重要。例如，商家通过收集大量对自身的评价，可以找出自己的产品、经营方面的优点和缺点，胜过费时费力的小样本用户调查；政府、组织通过了解网民发表的观点，可以更快捷地得知这些人群的关注点，便于体察民情，特别是一些国家在选举前，用这种方式可以较为客观地获知部分选情；此外，一些研究者还发现股票走势同社交媒体中的公众情绪有一定关联。公众在互联网上表现出的情绪特别具有宏观意义。

在 2012 年，受国际政治事件影响，中日关系出现波动，在我国互联网上则掀起了一阵阵“反日”、“抵制日货”的浪潮。在当年 9 月“九一八事变”纪念日来临之际，微博上的“抵制日货”相关话题显著增多，很多用户强烈地表达自己的愤慨之情。然而在往日（不受这些事件干扰的情况下），一些日本企业生产的商品由于性价比较高，很多人会对日货给予褒义评价。我们收集了当年 9 月与 10 月的一些与日本企业相关的微博，绘制出了一幅“差异化词云”，如图 7-4 所示。

图 7.4 2012 年 9 月与 10 月与日本企业相关微博的差异化词云，转绘自（崔安颀 2013）

图中黑色词汇是出现在 9 月份微博中的，白色词汇是在 10 月份中的。如果黑白混合（灰色），表明它在两个月份内均有出现，其偏黑（或偏白）的程度视比例而定。词的大小与其出现次数成正比。图中可见，9 月份微博包括“抵制日货”（上部偏右）、“中国人”（上部偏右）、“钓鱼岛”（上部偏左）、“日本车”（左侧）等民族主义情绪较为强烈的词汇（偏黑色的词汇）；而 10 月份因该事件的影响逐渐减弱，词汇转而反映一些抽奖活动，如“活动”、“玩转”、“参加了”等等（偏白色的词汇）。从这一分析中，我们可以清晰地看出公众关注热点的变化。

类似地，分别选择 2012 年 9 月、10 月这两个时间段，全国范围的褒贬倾向如图 7.5 所示。9 月份受到“抵制日货”相关贬义微博的影响，地图视图表现出浅色调；10 月份的公众情感则表现为中性，不同地区则可能呈现褒义（深色）、贬义（浅色）或接近中性（灰色）的情形，其各地区最大情感强度 0.006 小于 9 月份的 0.047，也说明公众的情感有所缓和。



地图数据源自 Google Chart Tools API，仅供示意。南海诸岛未画出。

图 7.5 2012 年 9 月（左）与 10 月（右）的全国日企相关微博公众情感分布，转绘自（崔安颀 2013）。浅色为贬义情绪，深色为褒义情绪

特别是在 9 月期间，褒义、贬义情感的变化趋势如图 7.6 所示，可以明显地看出贬义（深色）微博数量、程度在 9 月中旬达到高峰。在“九一八事变”纪念日临近时，网上的贬

义情绪非常强烈。随着政府的疏导和控制，这股浪潮才逐渐平息。

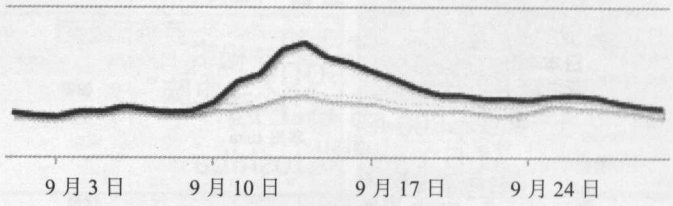


图 7.6 2012 年 9 月期间全国日企相关微博公众情感变化趋势，转绘自（崔安颀 2013）。深色为贬义情绪，浅色为褒义情绪

互联网上体现出的公众情感，有时是被一些影响力大的人物（称为意见领袖）所左右的；有时某些用户发表的特定观点、特定情感对组织和机构更有价值。图 7.7 展示的是 OrgSense 企业舆情分析演示系统的首页（Amiri, et al. 2012）。针对特定的一个组织、一家企业，该系统可以自动收集网上与之有关的微博，并分析出其中的关键用户，便于追踪和了解详情。

图 7.7 OrgSense 机构舆情分析演示系统

上面给出的案例是从宏观的视角分析情感，是整体层面的需求。对于用户来说，同样需要情感分析技术。例如，如果能便捷地检索到自己感兴趣的观点，就可以从海量的主观文本中找到最需要的信息，指导自己的消费行为。

以电子购物网站为例，这里汇聚了大量的用户评论信息。但通常购物网站的检索方式只

能按照某种好评率指标来排序筛选。对于复杂的商品，如果想检索某一种性能的指标，就需要人工浏览所有的评论。“京东商城”提供了“大家说”这一项筛选方式。如图 7.8 所示，针对“平板电脑”这种产品，从用户评论中自动找出产品特征（如“屏幕”、“待机时间”、“分辨率”、“电池”等）或用户需求（如“看电影”、“玩游戏”等），这样就可以细粒度地筛选特定功能的产品，便于用户选择。类似地，“亚马逊商城”也提供了按这类特征筛选商品评论的功能（如图 7.9 所示）。销售产品千差万别，显然人工总结每个产品的特征、性能是不现实的，然而用户会在评论里提及这些特性，这就使自动总结产品特征、评价成为了可能。

图 7.8 京东商城平板电脑类商品的筛选指标

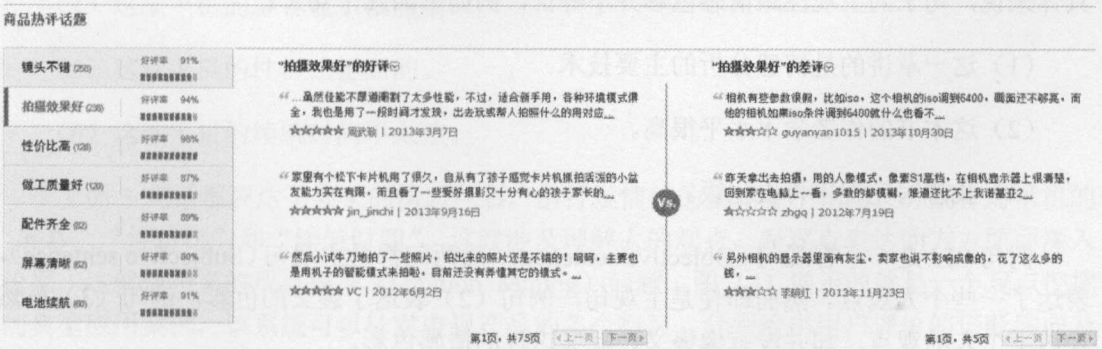


图 7.9 亚马逊商城某款相机的热评话题筛选

在大数据时代，用户产生数据成千上万，不可能人工逐条分析；同时，一项宏观的评价结果只有在大数据上才更有统计意义，更有说服力。这就是计算机科学的用武之地：尝试用计算机来处理这些海量的主观性文本，提取出文本中表达的情感和意见。

然而在很多人的头脑中，计算机就是一台冷冰冰的机器，计算机程序只是一些语句、符号的堆砌，算法原理无非是数学模型的应用。至于计算机程序员，更容易被认为是整天只跟机器打交道的，不善言辞、不善表达情感的操作工。那么计算机算法是如何了解人们的情感，读懂人们的心思呢？这就是这一章里我们将要讨论的内容。

本章首先列举一些这一领域里适合计算机解决的主要研究问题，然后对这些问题涉及的方法和技术予以讨论。接下来再介绍一些应用场景和案例，供读者朋友参考。

7.2 情感分析的主要研究问题

在介绍本领域相关的研究和应用问题之前，读者可能注意到本章标题“情感分析与意见挖掘”一共有两个组成部分：情感分析（Sentiment Analysis）和意见挖掘（Opinion Mining，又称观点挖掘）。狭义上看，似乎前者更偏重于分析喜怒哀乐这些情感，后者更偏重于理解用户表达的意见和观点。然而广义上，在研究界这两个词通常表示相同的研究内容。为了叙述方便，下文中以“情感分析”统一表示广义的研究领域。有兴趣的读者可以阅读文献（Pang & Lee 2008）的1.5节，来了解这两个词在学术研究界诞生和发展的历史。

提到文本的情感分析，我们最先想到的是判断一个句子有没有表达情感或观点。这个研究问题叫做主观性分析（Subjectivity Analysis）或主观性分类（Subjectivity Classification）。具体来说，句子的主观性跟情感色彩并不等价，例如下面几个句子：

（1）这一章讲的是情感分析的主要技术。

（2）这本书的作者学术水平很高。

（3）我想写完程序再去吃饭。

例句（1）是一个客观句（Objective sentence）；后两句都是主观句（Subjective sentence），表达了一些个人观点。然而即使是主观句，例句（2）表达了褒义的色彩，例句（3）虽然表达了个人的观点，却并没有像褒义、贬义这样的情感色彩。

接下来，判断文本的情感色彩也形成了一项具体的研究课题，即狭义的情感分析/分类

（Sentiment Analysis / Classification）。待识别的情感色彩都有哪些，则涉及具体的研究问题。例如最简单的可以认为是一项二分类问题：褒义或贬义，或者增加一类“中性”用于处理无情感色彩的句子。这样，情感分析就转化为二分类或三分类的分类问题，或者多步的二分类（第一步识别是否有情感，第二步针对有情感的句子，判断其是褒义还是贬义）。如果将情感色彩按照人的情绪来区分，例如“喜悦”、“愤怒”、“悲哀”、“恐惧”、“惊讶”等等，则可视作多分类，或者多个二分类（有无喜悦情绪、有无愤怒情绪等）问题来解决。当然，也可以将分类进一步细化，如图 7.10 所示，豆瓣网站将图书推荐分为 5 个级别，用 5 颗星星表示，1 星～5 星依次表示很差、较差、还行、推荐和力荐。这样体现出的情感程度更加具体。

图 7.10 豆瓣网站某本图书的评价

上面提到的都是分类任务，毕竟分类任务的定量指标比较适合数学模型的应用场景。这些任务的主要研究方法将在后文介绍。然而除了分类，人们还希望得到更详细的理解结果，例如：

（4）这款手机的屏幕显示很清晰。

（5）这款手机的性价比挺高的。

（6）这款手机的续航时间不太差。

上面 3 句话都表达了褒义的情感色彩，但抒发情感的主体各不相同，分别针对手机的“屏幕”、“性价比”和“续航时间”。这就涉及理解人的观点，对观点表达的方方面面深入挖掘，这就是狭义的观点挖掘（Opinion Mining）问题。图 7.11 展示的就是一个观点挖掘的典型应用案例，该系统可以针对电视产品的各个特征，分别给出用户评价的褒贬倾向及其程度。

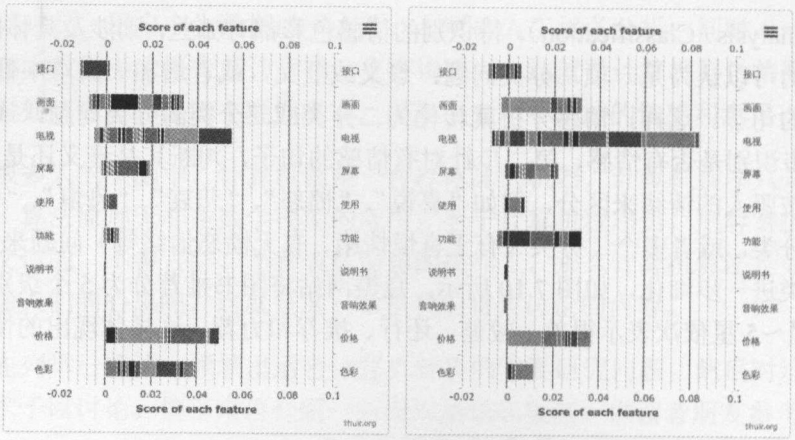


图 7.11 电视产品的产品特征挖掘与观点比较

看到这几个例子，有的读者可能已经体会到“观点”是由多个要素组成的，例如例句针对的事物是“手机”，抒发的观点具体到手机的“屏幕”、“性价比”等属性，具体的观点是“清晰”、“高”，等等。看来“挖掘”一词还过于笼统。在实际工作中，我们从原始文本中提取出观点的各个要素，这通常被称为观点抽取（Opinion Extraction），是观点挖掘的基础工作。抽取之后，我们还需要将抽取出的观点予以适当地组织，形成结构良好、便于应用的知识，作为一种情感资源，这就是情感资源构建（Sentiment Resource Construction）的主要目的。有了这些资源，用户便能够迅速查询到某一种观点的相关信息，这就是观点检索（Opinion Retrieval）。我们也可以认为它是信息检索（Information Retrieval）的一部分。

在情感、观点的分析基础上，针对不同的子问题和应用领域，又形成了很多具体的研究课题。例如：

- 垃圾观点识别（Opinion Spam Detection）由于我们分析所采用的互联网数据大都是用户提供的，自然就会存在虚假的、不真实的观点文本。例如商家可以雇佣“水军”（写手）为自己歌功颂德、对竞争对手口诛笔伐。因此需要对原始数据进行过滤，从而提供给用户更可信的结果；
- 情感摘要（Sentiment Summarization）传统的文本摘要（Text Summarization）是对长篇幅的文本提取出最有代表性、最能概括思想的一些句子，经过适当的连接修饰，形成短篇幅的摘要；情感摘要则侧重于提取出主要观点，使用户迅速理解原文的观点倾向；
- 舆情分析（Public Opinion Analysis）这是针对公众舆论进行的分析。舆情在商业领域又称口碑（Word-of-Mouth，WOM）。这个问题除了情感分析方面的技术，通

常还涉及其他研究内容，如主题检测跟踪（Topic Detection and Tracking，TDT）、趋势分析（Trending Analysis），以及社交网络分析（Social Network Analysis）领域常用的用户影响力分析（Social Influence Analysis）、信息扩散（Information Diffusion）技术，等等。

这些研究内容的层次更加深入，和具体的应用领域密切相关，也会涉及其他方面的技术，我们在本章就不做过多叙述了。如果读者朋友愿意深入了解，可以参看文献（Liu 2012）的相关内容。

7.3 情感分析的主要方法

7.3.1 构成情感和观点的基本元素

如何判断一句话是否表达情感，表达的情感是褒义还是贬义？让我们想想人是怎么判断的。例如前文例句（2）中，“很高”表达出了该句的情感；例句（4）的“清晰”一词也是对该款手机的一种肯定。可以看出，情感词是最常见的表达情感的元素。因此，倘若我们有一个很全面的情感词典，收录着各类情感词及其情感倾向、甚至情感程度，我们就可以按图索骥，在句中查找出现的情感词并将它们的情感倾向合并，即可得到一句话整体的情感倾向。后面将会介绍多种情感词典的构造方式。

除了汉字表示的词语，其他一些文本中的语言现象也可能表示情感。例如表 7.1 所示的表情符号、表 7.2 所示的表情图标，在网络文本中非常常见，更加形象生动地表达作者的情感。这是网络文本情感的重要语言现象，许多学者借此开展了深入的研究。

表 7.1 常见的表情符号及其含义

表情符号	含义
:-)	微笑
:-(	难过
:-D	大笑
:'-(	哭泣

表 7.2 常见的表情图标及其含义

表情图标	含义
	微笑
	委屈
	哈哈大笑
	哭泣

然而即便是相同的情感词，出现在不同的场合，可能表示截然相反的情感倾向。例如

前面提到的例句（5），性价比高是褒义倾向。但是：（7）这款手机的价格太高了。“高”一词用于产品价格时，则有贬义倾向。因此，一些情感词的情感倾向并不是始终如一的，需要结合具体的特征（属性）才能确定这一对特征词－观点词是褒义还是贬义。这在产品评论的分析中尤为明显。因此，构造“特征－观点对”也是观点挖掘中的一项基础工作。

看到这里，相信读者已经体会到了观点的组成要素。在学术研究界，学者们通常定义“观点”为如下要素组成的多元组。由于不同学者采用的要素名称不完全一致，以下列出主要的一些称呼。

- 观点的持有者（holder）：是表达这个观点的主体，如发表评论的作者、对一个事件表达同一反应的群体、发布报告的机构等。
- 观点对象或客体（target、entity、object）：是持有者所评论的、观点针对的对象，如某款产品、某个事件、某个人物等。
- 对象的属性、方面、特征（attribute、aspect、feature）：是对象的某个属性，如手机的屏幕、价格等。
- 表达观点的极性（orientation、polarity）：是指这个观点是褒义还是贬义，或者褒贬的程度是多少，表达哪种情绪等。

根据观点挖掘的应用场景不同，观点可能也需要包含如下要素。

- 观点的载体（carrier）：是指持有人发表的、承载观点的文章、评论等。
- 观点的时间（time）：有些观点的时效性较强，为分析观点随时间的变化状况，需要一并记录发表该观点的时间。

在观点抽取任务中，最关键的是自动识别和抽取对象的特征，并对其极性予以判断。我们在后文将介绍“属性－观点对”的分析方法。

此外，还有一类比较句，例如：

（8）我觉得中文的自然语言处理技术比英文技术更复杂。

（9）有人说基于规则的方法不如基于统计的方法准确度高。

这不光需要抽取出对象，还需要理解情感词是作用在哪个对象上，否则就会判断错误。通常我们要对句子进行更深入、更准确的句法分析，识别其中的多个命名实体及其关系。这是进一步的研究内容了，我们在本章不深入展开，读者可以阅读相关的文献予以了解。

至于反语、讽刺句，通常需要结合上下文理解，目前尚属困难的研究问题，我们也不做探讨了。

7.3.2 情感极性与情感词典

前面的示例已经介绍了常见的情感极性，即将人的情感划分为几种离散取值。常见的包括按照褒义、贬义以及中性来划分，也可以按照“喜怒哀乐”的情绪划分，如喜悦、愤怒、悲哀、恐惧、惊讶等（Xu, et al. 2010）。在心理学中，人的情感情绪还有多种模型来评价。例如：

- 美国心理学家罗伯特•普拉契克（Robert Plutchik）教授在 1980 年提出了“情绪轮”（Wheel of Emotions）模型（Plutchik 2001），如图 7.12 所示。这个模型包含了 8 种基本情绪：喜悦（joy）、信任（trust）、恐惧（fear）和惊讶（surprise）及其对立情绪：悲伤（sadness）、厌恶（disgust）、愤怒（anger）和期待（anticipation）。其中一些情绪两两组合还能形成其他情绪，例如喜爱（love）由喜悦与信任构成，其反面悲伤与厌恶则形成懊悔（remorse）。

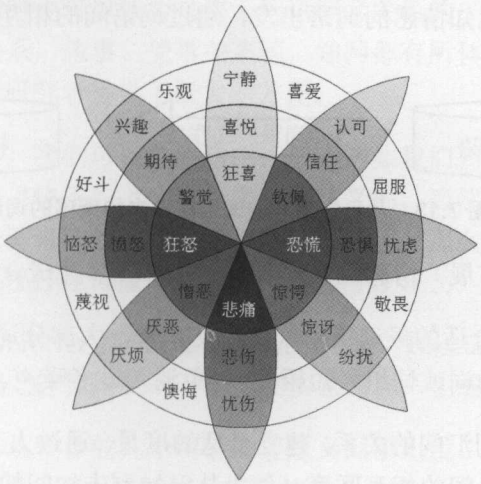


图 7.12 “情绪轮”模型

- 美国菲利普•谢弗（Phillip Shaver）教授与谢拉德•帕洛特（W. Gerrod Parrot）教授分别提出了情绪的层次模型（Shaver, et al. 1987, Parrot 2001）。在第一层次中，情绪分为喜爱（love）、喜悦（joy）、惊讶（surprise）、愤怒（anger）、悲伤（sadness）和恐惧（fear）6 种；每种情绪下分若干子情绪，例如喜爱包括爱慕（affection）、色欲（lust / sexual desire）与渴望（longing），所有子情绪共 25 个形成第二层次；类似地，第三层次又有 100 多种情绪。
- 美国心理学家查尔斯•奥斯古德（Charles E. Osgood）将人的情绪按效价（valence）、唤醒度（arousal）、优势度（dominance）来评价（Osgood 1952）。以此为模型，

美国佛罗里达大学几位学者整理了千余个英语单词所表达出的情绪程度，得到归一化的情感得分（Bradley & Lang 1999）；之后加拿大和比利时的学者在此基础上，进一步整理了近 14,000 个英语单词、词组的情感得分（Warriner, et al. 2013），还分析了不同性别、年龄、教育程度的人群对这些词语的情绪感受程度差别。

可见，当我们把人的情感用较为规范的若干类别来定义后，便可以整理得到各个词语究竟属于哪些情感类别。正如《新华字典》提供字义、《现代汉语词典》提供词义一样，情感词典提供给我们每个词的情感色彩，甚至情感程度。例如，“好”是褒义、“坏”是贬义；“手舞足蹈”比“喜形于色”的褒义程度更大。研究者构建并公开发布的情感词典，虽然规模不大，但可信度高，可以作为情感分析的第一步工具。

然而，词典收录的词条毕竟有限，人工构建费时费力且不现实。那么有没有办法用计算机算法识别词条的情感倾向呢？进一步说，词典收录的词条总是固定的，网络新词那么多（如“给力”、“喜大普奔”等），我们有什么办法能自动扩展词典呢？我们所采用的方法就是从已知到未知：从已知情感的词语出发，通过词语间的相互联系，探索未知情感的词语，如图 7.13 所示。

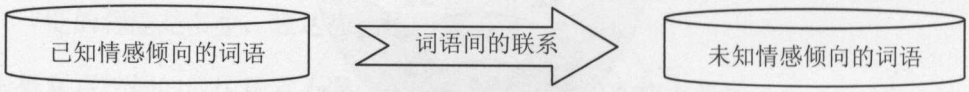


图 7.13 从已知情感的词语探索未知情感的词语

因此，自动构建（扩展）情感词典可以通过如下三步完成。

1. 确定目标：选取合适的候选情感词。在文本中，大部分词语并没有情感倾向。我们需要找出能够表达情感的词或词组（短语），以免混入过多噪声。

2. 建立桥梁：通过词汇间的关系，建立合适的度量。通过人工或自动构建的词义网络，我们可以得到候选词语之间的相互联系，作为从已知到未知间的桥梁。

3. 传播情感倾向：确定了候选词，相当于架起了桥墩；建立了度量，相当于搭上了桥面；接下来就可以采用适当的方法，从已知情感倾向的词语出发，把它们的情感信息传播出去，使每个未知的候选情感词都被赋予一定的情感倾向。从数学角度而言，倾向性可以用得分（情感得分）来表示，这便可以用数学模型来计算情感倾向了。

这三步之间也是相互联系的。例如，如果目标广（噪声多），选取的度量就需要比较准确，否则错误的信息也会被传播出去，造成干扰；如果候选词语含有很多新词，那么采用的词语关系度量也应当能覆盖这些传统词典未收录的词汇，否则新词就成为了孤立点，无从联系。以下我们首先从第 2 步出发，针对常见的几种词汇关系进行介绍。

方法一：词义关系

在文本中，词语的情感倾向取决于词义。因此，对于已知情感倾向的词语，它们的近义词应该具有相似的情感倾向、反义词应该具有相反的情感倾向。这样，寻找未知词语情感倾向问题，就转化为寻找相似词语的问题。

在自然语言处理研究界，已有许多学者对词汇的含义、关系进行了梳理，并建立了词义网络。常用的中英文词汇知识库包括如下。

- 英文资源：普林斯顿大学研发的 WordNet。它不但收录了大量的词条，而且针对各个词语的不同词性、不同含义，按照相同含义进行分组，形成同义词集合（synset）。不同的同义词集合间也建立了联系。这样，所有词汇形成了一个网络，可以便捷地查询到一个词语某个含义的近义词、上下层蕴涵关系，等等。
- 中文资源：由董振东、董强等学者研发的知网（HowNet）。这也是一个词语含义的关系网络，同时也可以认为是揭示词语概念关系的一个知识网络。不同概念间的同义、反义关系，施事、受事关系等，知网都有所体现。知网还有词条相对应的英文释义，为研究者提供了极大的方便。

有了近义词关系网络，我们可以扩展情感词集。例如我们初始知道“高兴”具有褒义情感，那么它的近义词如“快乐”、“欣喜”等都是褒义的。当然，这样描述比较粗糙，特别是经多次传递后，两个词语间的含义可能不那么接近了，那么其相互影响就应当减弱。利用一些数学模型，可以较好地传播情感倾向（情感得分）。

值得一提的是 SentiWordNet（Esuli & Sebastiani 2006，Baccianella, et al. 2010）。顾名思义，这是在 WordNet 基础上计算得到的一个英文情感词典，生成词典的步骤如下。

1. 选取种子词集合：从较少的几个明确的褒贬义词语出发，通过 WordNet 里定义的二元关系（如相同极性的“also－see”或相反极性的“direct antonymy”等关系），扩展这些词语（可控制传播半径），形成褒义和贬义的词语集合，作为“种子”集合。

2. 训练分类器：用上述的褒贬种子词集，再加上一个中性种子词集一起，作为训练数据，即训练一个三分类器。由于 WordNet 定义了同义词集（synset），这个分类器训练的单位不再针对一个个词语，而是针对一个个同义词集。

3. 标记其他同义词集：利用上一步得到的分类器，可以对 WordNet 中所有的同义词集进行标记，得到它们的情感倾向。

4. 研究者发现，最佳的分类效果需要训练多个分类器，分别采用不同的传播半径和分类模型，最后由多个分类器投票决定待标记词集的情感倾向。

5. 采用随机游走（random walk）模型分别对得到的褒义词集、贬义词集的情感倾向进行调整。当游走过程收敛后，即得到最终结果。

其他一些利用 WordNet 的工作方法也类似。例如（Kim & Hovy 2004，Hu & Liu 2004），都是选取一些已知情感倾向的动词或形容词作为种子词，利用同义词关系扩展这个集合，或计算与之在 WordNet 中的共现概率。这样，在 WordNet 中出现的其他词语，只要被同义词关系覆盖到，都可以计算出一定的情感倾向。

方法二：句法关系

词义关系毕竟是依靠已总结出的关系网络。但在很多时候，一些词语并不能被网络覆盖；还有些词语的用法、词义比较灵活，现有的知识库尚未涵盖。这就需要寻找其他关系。

我们可以利用句子信息来建立词语间的桥梁。在较长的句子中，不同分句间往往有连词作连接。并列连词连接着并列含义的句子；转折连词连接相反含义的句子。因此，两个分句中的词语就可以用连词信息来判断是相近含义还是相反含义。如果我们的语料足够大，那么这些词汇在分句间的联系就有了一定的统计意义，可以作为比较可信的关系度量。

在（Hatzivassiloglou, et al. 1997）工作中，作者采用两千多万篇新闻语料。选取形容词作为候选词语，进而利用“and”、“or”、“but”等连词构建词语间的相互联系。这些近义或反义的词对形成了一张图（graph），接下来可以采用优化或聚类的方式，将相近倾向的词语划分为一个个簇；再根据簇内已知情感倾向词语的多少，来标记整个簇内词语的情感倾向。

这种句子间的关系，比起前面的词义关系来说要弱了许多，会有更多的噪声。但对于大的数据规模而言，真实的词语关系会多次出现在语料中；不真实的词语关系出现次数较少。因此，这个方法可以在大数据下得到较好的应用。

但如果我们的句子连词较少，各个句子之间的联系并不紧密，我们还能找到词语间关系吗？

方法三：同现关系

研究者们注意到了这样一个现象：表示相同情感倾向的词语更可能共同出现，但相反倾向的词语则较少共同出现（Beineke, et al. 2004）。在较大尺度的数据中，这个现象更加突出。在大数据时代，语料相对是充足的，因此利用“共同出现”——即“同现”这个特点便可以计算词语的情感倾向。

可以看出，同现关系比前面两者更弱，噪声更多。因此，一方面我们可以增强同现关系的可靠性；另一方面我们可以在倾向性得分传播时，予以过滤，以滤除噪声。

对于增强同现关系的方法，通常的做法是扩大语料规模，使同现关系更可靠。但是语料扩大会导致计算能力需求的攀升，而且并不是所有噪声都能被降低。因此，还可以采取的方法如下。

- 选取一个适当的“窗口”，即一个词的上下文中，相邻 k 个以内的词才算是同现；或者按照词间距离，将其同现的权重逐渐衰减——这其实表明同现关系不能无边无沿。
- 采用适当的方法衡量同现关系。我们不能仅仅根据同现次数的多少来认为两个词之间是有联系的。例如“的”字频繁地跟很多词同现，但这并不能说明其独特性。这时除了用概率估计外，还可以采用“点对互信息”（Pointwise Mutual Information，PMI）来衡量两个词（或词组）之间的相关性：

$$PMI(word_1, word_2) = \log(P(word_1, word_2) / P(word_1) / P(word_2))$$

例如文献（Turney 2002）中利用下式来计算一个短语的褒贬程度：

$$Polarity(phrase) = PMI(phrase, \text{“excellent”}) - PMI(phrase, \text{“poor”})$$

这表示短语的情感极性是它跟“好”词相关程度与它跟“坏”词相关程度之差。

利用“词嵌入”（word embedding）方法，将每个词用多维向量表示。通过学习算法，可以使得相似含义的词语向量距离较近，从而发现相似词语。这一方法把成千上万的词汇（高维空间）转化为低维向量表示，便于计算和挖掘词义。Google 公司的研究员们提出的 Word2vec 算法是一个经典的例子，可以改善情感分析任务的性能。

对于过滤得分传播的方法，研究者们通常选用适当的算法进行传播计算。

考虑同现关系将词语组成的一个个词对可以构造一个图。之后可以仿照前面的做法来计算候选词语的情感倾向。这里选用的算法需要能对强的关系予以增强、弱的关系予以减弱。例如文献（Serban, et al. 2012）中将子图中的完全图（clique）作为强关系；文献（Banea, et al. 2008）在每次迭代计算新候选词的情感得分后，采用相似性度量进行过滤，只保留与原始种子集最相似的新词集合。

以上是常见的几种词语度量关系。至于候选情感词的选取，通常和应用问题或语料密切相关。例如上文也提到了，传统文本中通常只采用形容词、动词等词语作为候选情感词。然而在很多评论文本中，一些短语也可能具有整体的情感倾向。文献（Turney 2002）中即定义了若干规则来通过特定词性的词生成候选的情感词组，如表 7.3 所示。

表 7.3 表达情感的短语构成规则（Turney 2002）

第 1 个词	第 2 个词	第 3 个词	短语示例
形容词	名词	（任意）	online experience，在线的体验
副词	形容词	非名词	very handy，非常方便
形容词	形容词	非名词	（多为句子片段）
名词	形容词	非名词	programs such（句子片段）
副词	动词	（任意）	probably wondering，可能想知道

此外，本节开始提到在社交网络类文本中，情感符号也是常见的表达情感倾向的元素。因此情感符号也可以作为情感候选词的一类（崔安颀 2013）。尤其是在微博等互联网新媒体文本中，各式情感符号、情感图标层出不穷，而这些“词”并不为传统词典所涵盖。因此采用前述的基于同现的分析方法可以自动计算得到情感符号的情感倾向，从而帮助对这类新媒体文本进行情感分析。

对于情感得分的传播，由于词义关系构成了图，自然可以采用多种基于图的算法进行计算。除了前文提到的图内聚类方法，常用的算法还有图传播（graph propagation）（Velikovich, et al. 2010）与标签传播（label propagation）（Zhu & Ghahramani 2002）等。这两者的基本思想都是从若干种子词出发，每次更新与已有得分词相连的新词得分，使情感得分逐渐传播到全图。两者的区别在于：在标签传播算法中，未知得分词语受到所有与之相连的已知得分词语影响；而在图传播算法中，未知词只受到关系最强的一个已知词影响。如果全图中有许多比较密集的子图，或者图的质量不高，那么采用标签传播算法就会使这些子图涉及的词语受到很强的噪声干扰。因此，需要根据语料、度量方式，选择适当的传播算法。

7.3.3 属性－观点对

在观点挖掘任务中，观点的倾向性只是其中一个要素。观点的对象、属性同样重要。例如在舆情分析这一类应用中，仅仅知道观点的褒贬是不够的。如购物网站，一款手机的总体评价只能给人一个粗略的判断，但尚不足以影响人的行为。消费者、商家更需要知道手机的哪个功能是好是坏、是否符合自己需求。因此，挖掘观点所针对的对象属性是一项很有意义的研究工作。类似前文识别情感词的步骤，我们也可以从语料中提取出属性词和与之对应的观点词。

前文提到，情感词之间有语义、句法等层次的联系。通过这种联系，我们可以从已知情感词出发，探索到新的情感词。现在我们面对的是属性词和观点词（即修饰属性的情感

词），那么这两者之间有没有什么联系呢？实际上在多数情况下，属性词和观点词是出现在同一个句子中的，这就是一个自然的同现关系。通过这种联系，我们就可以利用传播的方式发现属性－观点对。

值得一提的是确定候选的属性词和观点词。前面我们介绍了采用词性来筛选候选词的一些规则，这里同样我们可以将一句话中的形容词作为候选观点词、而名词（或名词组合等）作为候选属性词。例如文献（李智超 2011）以如下规则作为抽取属性词的模式，模式里出现的“动词”代表同一词语的名词含义，但被词性标注标记为动词的情形，如表 7.4 所示。

表 7.4 模式匹配抽取属性词的示例（李智超 2011）

模式	示例
名词	价格；电池；画质
名词+名词	快门/速度；照片/效果
动词	设计；成像；摄影
动词+名词	续航/能力；成像/效果
缩略语	性价比；质保

这一利用规则寻找候选属性词的前提是分词算法能正确地将该词语识别出来，并赋予正确的词性。但有一些传统词典未收录的词（称为“未登录词”），可能会被划分为多个字、词。例如提到数码相机时常用的“防抖”一词，是指相机的一项属性，但传统分词算法通常会将其识别为一个动宾短语并分为两个词“防/抖”。这时我们可以利用一些新词发现的方法，特别是对于大规模的语料，采用基于统计的办法来识别。以文献（李智超 2011）中采用的“上下文熵”（Context Entropy）方法为例，如图 7.14、图 7.15 所示。

图 7.14 数码相机领域评论语料中“抖”字左侧上下文示例（李智超 2011）

图 7.15 数码相机领域评论语料中“防抖”的左侧上下文示例（李智超 2011）

在图 7.14 中，“抖”字左侧的字比较集中，大部分都是“防”字，因此“抖”字可向左扩展，吸收“防”字形成一个完整的（可能）词语“防抖”。但在图 7.15 中，“防抖”的左侧则并无某一集中出现的字，说明“防”字左侧不应继续扩展，已到达词语边界。采用这个方法，可以较为全面地提取出候选的属性词。

当然，这样提取出的候选属性词、候选观点词仍然较为粗糙。可以再通过一些语法规则或背景语料进行过滤。例如，连词“和”、“与”左右两侧连接的词语（名词），如果其中一个是已知的属性词（或观点词），那么另一个便更可能是一个属性词（或观点词）；紧挨着已知属性词之后的形容词更可能是观点词；根据句法分析得到的依存关系树（Dependency Tree），可以提取包含已知属性词（或观点词）的主谓关系结构（Subject-Verb，SBV）中的另一成分作为候选观点词（或属性词）；或者从大量的与现有语料无关的背景语料中，统计出现频率最高的词语（称为“高频词”），这些词不太可能是与领域相关的属性词（李智超 2011，翟忠武 2011）。在这一过程中我们看到，给定一个已知的属性词或观点词列表，可以逐步（迭代地）扩展生成更多的属性词或观点词。这与我们之前的情感倾向计算方法也有相似之处。

现在我们得到了候选的观点词、属性词，利用句中（上下文）的同现关系将它们构成了“属性－观点”对。这个观点词、属性词网络与之前的情感词网络十分相似，只不过这里的结点不只是单一的一种情感词，而是有两类。我们以每一对属性－观点对作为一个基本单元，考察它们在上下文间的关系，是由并列连词还是转折连词相连接，这就构建了一个新的图。之后便可用与前面相似的算法进行迭代与传播，直到把每个属性－观点对都赋予一定的情感得分。

采用“对”作为整体的原因，是同一个观点词针对不同的属性词，其情感倾向可能不同，如前文所述的“价格－高”与“性价比－高”。当我们将每个“对”作为结点，它们便不再受单一的倾向所制约。

7.3.4 情感分析

我们构建情感词典、属性－观点对这些情感资源，是为了对一篇文本进行情感分析，即识别该文本表达了褒义、贬义等哪种情绪，其情感的强烈程度是多少。最基本、也是最常见的是情感分类任务：识别一段文本的情感是褒义、贬义还是中性。作为一项文本分类任务，通常有无监督学习（Unsupervised Learning）和有监督学习（Supervised Learning）两种方式。下面我们就依次介绍无监督学习、有监督学习的情感分析方法。

1. 无监督学习

句子的情感信息完全来源于其中的情感词（或涉及的观点词等）及其在句中的地位。

以无监督学习方式对文本进行情感分类，由于无需训练语料，我们需要人工找出情感词与句子情感之间的联系。

最直接的想法是在一句话中对出现的情感词、属性－观点对进行匹配，并将匹配到的各个情感倾向得分相累加，得到整句话、整段的总体情感倾向。例如：

这个手机很漂亮，价格也便宜，就是电池发热太严重了。

这句话中涉及的情感词语有“漂亮”、“价格－便宜”、“发热－严重”，假定它们的情感分数分别是 0.8、0.3、-0.7（正分表示褒义、负分表示贬义）。因此整句话的情感得分为：0.8+0.3-0.7=0.4，句子总体表达了褒义的情感。只要我们能保证情感词典等资源的情感信息质量较高（比较准确），这种方式简单快捷，特别是对于一些短文本、句式简单的文本，不失为一种解决方案（Cui, et al. 2011）。

但这样得到的结果一定正确吗？如下面的三个例句：

（1）这种情感分析方法很成功。

（2）这种情感分析方法不很成功。

（3）这种情感分析方法很不成功。

三个例句中都有褒义的“成功”一词，但因为否定副词“不”的出现，使后两句表达的情感倾向出现反转，表达贬义倾向。进一步地，其中程度副词“很”的位置差异，还会影响情感的程度：例句（2）中的“不很”表达较弱的否定，而例句（3）中的“很不”则是强烈的否定。因此，我们需要对句子中的否定副词、程度副词等成分加以提取和分析。如例句所示，通常的做法是寻找邻近的否定副词、程度副词、情感词，并根据它们之间的排列顺序，用程度副词作为情感词的加权系数、用否定副词作为反转系数。假定“成功”一词的情感得分为 0.8（正分表示褒义），“很”字的加权系数为 1.2，那么

（1）程度副词+情感词：情感得分为 1.2×0.8=0.96。

（2）否定副词+程度副词+情感词：情感得分为(-1)×(1/1.2)×0.8=-0.67。

（3）程度副词+否定副词+情感词：情感得分为(-1)×1.2×0.8=-0.96。

注意“不很”的否定效果比单独用“不”字要弱，因此对程度副词的加权效果取倒数处理。

与之类似，多个句子之间可能也有起衔接作用的词语，这将影响几个句子组成的篇章

的情感倾向。例如：

（4）这个电影的演员是一流的，画面也不错，就是剧情太烂了！

如果统计褒贬情感词语的个数，褒义词（“一流”、“不错”）有2个，贬义词（“烂”）只有1个。然而整句话读下来，相信读者会感受到，作者是把重点放在最后一句，批评剧情。因此，如果我们仅仅采用一些简单的词汇搭配规则，容易出现错误。特别是对较长的句子或篇章，准确率将会降低。

如果能够用数学模型自动从大规模的训练语料中表达（学习）出词汇、语句、篇章的一些内在规律，相信会比我们人工逐条总结得到的规律会有效得多。因此，我们可以采用有监督的机器学习模型来完成情感分析任务。

2. 有监督学习

（1）训练数据

进行有监督学习的一项必不可少的环节就是训练数据，即已经标注出答案的语料。这样，通过训练，模型可以建立起特征与结果类别之间的分布关系，从而对未知答案的数据也能给出其类别的估计。在情感分析任务中，互联网大数据是一个很好的训练数据来源。本章开始提到的产品评论网站、餐馆评价网站、书评影评网站，都汇聚了大量的情感分析语料。最重要的是，这些网站为了让数据处理更加方便，在用户提交文字评论的同时，通常还有一个打分项，即用1～5或给出星级的方式让用户给出一个总体评价。从机器学习的角度来看，这个项目即可作为对应文字评论的情感标注。这样，我们就可以获得充足的训练语料。

如果不是这种评论类文本、网站，例如对于微博文本，我们如何获得训练数据呢？人工标注固然是一种解决方案，但费时费力。而且情感本身比较主观，因此对标注员的数量、质量都有一定要求。针对这一现象，有研究者注意到一些互联网文本（特别是微博）中有表情符号，揭示了作者的心情。而且这些符号是作者写的，可靠程度高。因此，可以挑选出一些简单的、包含表情符号的文本（复杂句子可能影响因素过多），用这些表情符号作为标注。当然，采用这样的标注原则形成训练数据，未免噪声较多。在这种“远监督”（distant supervision）的框架下，有噪声标注数据在用于训练时需要经过多重处理，逐步去粗取精，最终得到优质分类器（Go, et al. 2009, Pak & Paroubek 2010）。

（2）特征空间

获得足够的训练数据后，我们还需要从文本中提取出特征，将每个文本映射到一个向量。在这样的空间下，机器学习模型才能发挥作用。在此，我们综合多位研究者的工作，

列举常见的与情感相关的文本特征，供读者参考。

- 基于 *n*-gram 的词袋（bag-of-words）模型，通常选取 n=1～3，即 unigram、bigram 和 trigram。
- 基于分词的词袋模型，亦可以参照 *n*-gram 的做法，以词为基本单位（unigram），并形成 bigram、trigram 等。
- 出现的情感词，在给定情感词典的情况下，情感词更有可能揭示整段文本的情感倾向，因此将命中的情感词作为向量的维度，有助于模型学习。这里情感词可能包括正规词语、表情符号等。
- 词性特征，可以反映文本中各个词性的分布。
- 副词类特征，包括否定词、最高级及比较级类词汇等。
- 词汇的扩展。可以对出现的词语按词义进行扩展，如同义词、反义词等，从而增加命中特定词汇的机会。
- 句法特征。如词语在句法分析树中的父亲结点词，在句中的地位（主语、宾语等），以及是否是连接词等。
- 其他符号：
  - 标点符号：问号、感叹号的出现往往表示有较强的情感，而分号、顿号通常用于大段排比中，有可能是客观句；
  - 百分号、数字编号：较多的百分号、数字编号可能是在罗列一串条目，较为书面和中立；
  - URL（网址）：对于短句，如果有 URL，通常是陈述或广告，因此其更有可能是客观句。

在计算权重时，除了采用词的次数之外，还可以采用比例、对数、tf-idf 等加权方式进行调整。

虽然上述特征列出的较多，但在实际工作中可以采用一些特征选择的方法予以筛选，或采用压缩（compression）、提取特征值（eigenvalue）、矩阵奇异值分解（Singular Value Decomposition，SVD）等方法，减小向量空间的维度，从而在一定程度上缓解输入过于稀疏的问题。

（3）学习模型

在情感分析中，常见的分类器及其应用方法包括如下几种：

- 朴素贝叶斯（Naive Bayes）：分类的类别集合为 C = {褒义, 贬义, 中性}，假设各个特征之间相互独立，则给定文本 S，它属于类别 c_i 的概率 $p(c_i|S) \propto \prod p(w_j|c_i)$ ，

其中 $p(w_j|c_i)$为训练样例中类别为 c_i 的数据中特征 w_j 出现的概率。最终 S 的所属分类取 $p(c_i|S)$较大的那个所对应的 c_i。

- k 近邻（k-Nearest Neighbors，kNN）：给定文本 S，首先选出 S 与训练样例中最近的 k 个数据点，对这 k 个点的倾向性，按其与 S 的距离倒数加权，并求和作为 $p(c_i|S)$。
- 支持向量机（Support Vector Machine，SVM）：在核函数的作用下，该模型将向量投射到超空间中的支持向量，并寻找最优的一个划分的超平面，使支持向量到这个超平面的距离最大。
- 最大熵模型（Max Entropy）：在满足约束条件的前提下，使熵值 $-\sum p(c_i|S) \cdot \log p(c_i|S)$ 达到最大。因此 $p(c_i|S) = Z(S)^{-1} \exp(\sum \lambda_j \cdot w_j)$。

此外，如果将情感词的识别作为一个结构化标注问题来看待，还可以采用隐马尔科夫模型（Hidden Markov Model，HMM）（Jin & Ho 2009）、条件随机场模型（Conditional Random Field，CRF）（李方涛 2011）等，详见相关文献。

7.4 主要的情感词典资源

在前面章节的叙述中，已经穿插介绍了一些公开的情感词典。这里我们整理如下，方便参考。

中文领域有如下几个情感词典（见表 7.5）。

表 7.5 中文公共情感词典及其词语条目数

词典	褒义（正面倾向性）词数	贬义（负面倾向性）词数
《学生褒贬义词典》（张伟，等 2004）	728	933
知网“情感分析用词语集”	836	1,254
台湾大学情感词典（Ku & Chen 2007）	2,810	8,276
清华大学构建的情感词词典（Li & Sun 2007）	5,567	4,468

此外，还有北京大学研发的情绪词典（Xu, et al. 2010），包括喜悦情绪的词条 91 个，愤怒的词条 112 个，悲哀的词条 89 个，恐惧的词条 103 个，以及惊讶情绪的词条 92 个。虽然数量较少，但这种情绪划分方式比褒贬情感更细致，对一些针对细粒度的情绪分析很有帮助。

英文领域的情感词典还包括如下几项：

- SentiWordNet。基于 WordNet 网络计算得到的情感词典，优势在于同一词语的不

同释义可能得到不同的情感得分；

- LIWC（Linguistic Inquiry and Word Count）。这是由美国德州大学奥斯汀分校、新西兰奥克兰大学的几位研究者开发的一套软件，也包含多语言的情感词典。对词语的区分很细，包括心理特性（情感词、认知词等）、个人化（工作、休闲等）等不同维度的标注。但使用需要收费；
- ANEW（Affective norms for English words）（Bradley & Lang 1999）。千余个英语单词的归一化情感，按照效价、唤醒度、优势度这三个维度进行评价；
- MPQA（Multi-Perspective Question Answering）。由美国匹兹堡大学研发的若干情感语料（英文），对情感词标记了词性、情感倾向、情感强度等信息。

随着时代的发展、语境的变迁，词典收录词条的情感倾向也可能发生变化。因此在实际应用中还需要结合现实背景和应用需求，来选择合适的词典。

7.5 内容回顾与推荐阅读

在本章中，我们介绍了什么是情感分析和意见挖掘，它的重要意义和应用价值在哪里；我们了解了情感分析的主要研究内容，并且一步步地熟悉了完成情感分析和意见挖掘的基本流程。限于篇幅，大部分方法没有详细展开，数学模型、计算方法的细节没有一一列出。有兴趣的读者可以阅读相应的文献。

在本章开始，我们提到了“星期一综合征”。在大数据时代，通过对网民发表的文本进行分析，我们就可以了解这些群体的整体情感倾向。笔者分析了 2010 年初的一百万条新浪微博数据，绘制成了图 7.16，供读者参考。你们看，网友的心情变化趋势是不是像漫画中画的那样呢？

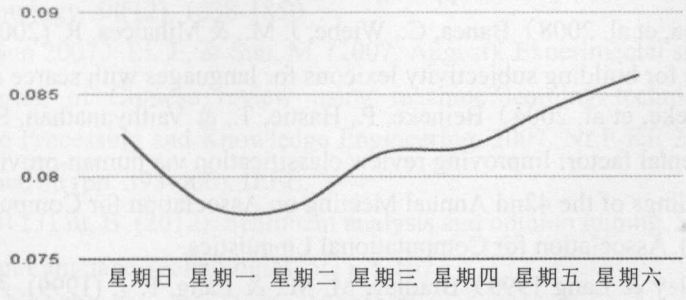


图 7.16 一周微博情感变化。纵轴数值表示褒义倾向程度，数值越高表示褒义程度越强。

看到这里，读者一定能体会到情感分析在互联网时代的重要价值。我们也看到，这方面的研究仍然有很广阔的拓展空间。希望这一章能给读者带来一些启发，使更多的人能够参与到这方面的研究和应用中来，使计算机真正能读懂人们的心。

以下是一些推荐阅读的文献：

- （Liu 2010）Liu, B. (2010). Sentiment analysis and subjectivity. Handbook of natural language processing, 2, 627-666.
- （Liu 2012）Liu, B. (2012). Sentiment analysis and opinion mining. Synthesis Lectures on Human Language Technologies, 5(1), 1-167.
- （Pang & Lee 2008）Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and trends in information retrieval, 2(1-2), 1-135.
- （Turney 2002）Turney, P. D. (2002, July). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In Proceedings of the 40th annual meeting on association for computational linguistics (pp. 417-424). Association for Computational Linguistics.

7.6 参考文献

[1] （Amiri, et al. 2012）Amiri, H., & Chua, T. S. (2012, October). Mining sentiment terminology through time. In Proceedings of the 21st ACM international conference on Information and knowledge management (pp. 2060-2064). ACM.

[2] （Baccianella, et al. 2010）Baccianella, S., Esuli, A., & Sebastiani, F. (2010, May). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In LREC (Vol. 10, pp. 2200-2204).

[3] （Banea, et al. 2008）Banea, C., Wiebe, J. M., & Mihalcea, R. (2008). A bootstrapping method for building subjectivity lexicons for languages with scarce resources.

[4] （Beineke, et al. 2004）Beineke, P., Hastie, T., & Vaithyanathan, S. (2004, July). The sentimental factor: Improving review classification via human-provided information. In Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics (p. 263). Association for Computational Linguistics.

[5] （Bradley & Lang 1999）Bradley, M. M., & Lang, P. J. (1999). Affective norms for English words (ANEW): Instruction manual and affective ratings (pp. 1-45). Technical

Report C-1, The Center for Research in Psychophysiology, University of Florida.

[6] （Cui, et al. 2011）Cui, A., Zhang, M., Liu, Y., & Ma, S. (2011). Emotion tokens: Bridging the gap among multilingual twitter sentiment analysis. In Information retrieval technology (pp. 238-249). Springer Berlin Heidelberg.

[7] （Esuli & Sebastiani 2006）Esuli, A., & Sebastiani, F. (2006, May). Sentiwordnet: A publicly available lexical resource for opinion mining. In Proceedings of LREC (Vol. 6, pp. 417-422).

[8] （Go, et al. 2009）Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 1, 12.

[9] （Hatzivassiloglou, et al. 1997）Hatzivassiloglou, V., & McKeown, K. R. (1997, July). Predicting the semantic orientation of adjectives. In Proceedings of the 35th annual meeting of the association for computational linguistics and eighth conference of the european chapter of the association for computational linguistics (pp. 174-181). Association for Computational Linguistics.

[10] （Hu & Liu 2004）Hu, M., & Liu, B. (2004, August). Mining and summarizing customer reviews. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 168-177). ACM.

[11] （Jin & Ho 2009）Jin, W., Ho, H. H., & Srihari, R. K. (2009, June). A novel lexicalized HMM-based learning framework for web opinion mining. In Proceedings of the 26th Annual International Conference on Machine Learning (pp. 465-472).

[12] （Kim & Hovy 2004）Kim, S. M., & Hovy, E. (2004, August). Determining the sentiment of opinions. In Proceedings of the 20th international conference on Computational Linguistics (p. 1367). Association for Computational Linguistics.

[13] （Ku & Chen 2007）Ku, L. W., & Chen, H. H. (2007). Mining opinions from the Web: Beyond relevance retrieval. Journal of the American Society for Information Science and Technology, 58(12), 1838-1850.

[14] （Li & Sun 2007）Li, J., & Sun, M. (2007, August). Experimental study on sentiment classification of Chinese review using machine learning techniques. In Natural Language Processing and Knowledge Engineering, 2007. NLP-KE 2007. International Conference on (pp. 393-400). IEEE.

[15] （Liu 2012）Liu, B. (2012). Sentiment analysis and opinion mining. Synthesis Lectures on Human Language Technologies, 5(1), 1-167.

[16] （Osgood 1952）Osgood, C. E. (1952). The nature and measurement of meaning. Psychological bulletin, 49(3), 197.

[17] （Pak & Paroubek 2010）Pak, A., & Paroubek, P. (2010, May). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In LREC (Vol. 10, pp. 1320-1326).

[18] （Pang & Lee 2008）Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and trends in information retrieval, 2(1-2), 1-135.

[19] （Parrot 2001）Parrott, W. G. (2001). Emotions in social psychology: Essential readings. Psychology Press.

[20] （Plutchik 2001）Plutchik, R. (2001). The nature of emotions. American Scientist, 89.4, 344-350.

[21] （Serban, et al. 2012）Serban, O., Pauchet, A., Rogozan, A., Pécuchet, J. P., & LITIS, I. Semantic propagation on contextonyms using sentiwordnet. In WACAI 2012 Workshop Affect, Compagnon Artificiel, Interaction (p. 86).

[22] （Shaver, et al. 1987）Shaver, P., Schwartz, J., Kirson, D., & O'connor, C. (1987). Emotion knowledge: further exploration of a prototype approach. Journal of personality and social psychology, 52(6), 1061.

[23] （Turney 2002）Turney, P. D. (2002, July). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In Proceedings of the 40th annual meeting on association for computational linguistics (pp. 417-424). Association for Computational Linguistics.

[24] （Velikovich, et al. 2010）Velikovich, L., Blair-Goldensohn, S., Hannan, K., & McDonald, R. (2010, June). The viability of web-derived polarity lexicons. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (pp. 777-785). Association for Computational Linguistics.

[25] （Warriner, et al. 2013）Warriner, A. B., Kuperman, V., & Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. Behavior research methods, 45(4), 1191-1207.

[26] （Xu, et al. 2010）Xu, G., Meng, X., & Wang, H. (2010, August). Build Chinese emotion lexicons using a graph-based algorithm and multiple resources. In Proceedings of the 23rd International Conference on Computational Linguistics (pp. 1209-1217). Association for Computational Linguistics.

[27] （Zhu & Ghahramani 2002）Zhu, X., & Ghahramani, Z. (2002). Learning from labeled and unlabeled data with label propagation. Technical Report CMU-CALD-02-107, Carnegie Mellon University.

[28] （CNNIC 2014）中国互联网络信息中心，中国互联网络发展状况统计报告（2014

年 1 月）。

[29]（崔安颀 2013）崔安颀. 微博热点事件的公众情感分析研究 [博士学位论文]. 北京：清华大学计算机科学与技术系，2013.

[30]（李方涛 2011）李方涛. 基于产品评论的情感分析研究 [博士学位论文]. 北京：清华大学计算机科学与技术系，2011.

[31]（李智超 2011）李智超. 面向互联网评论的情感资源构建及应用研究 [博士学位论文]. 北京：清华大学计算机科学与技术系，2011.

[32]（谢丽星 2011）谢丽星. 基于 SVM 的中文微博情感分析的研究 [硕士学位论文]. 北京：清华大学计算机科学与技术系，2011.

[33]（翟忠武 2011）翟忠武. 网络舆情分析方法研究 [博士学位论文]. 北京：清华大学计算机科学与技术系，2011.

[34]（张伟，等 2004）张伟，刘缙，郭先珍. 学生褒贬义词典. 北京：中国大百科全书出版社，2004.

第 8 章

面向社会媒体大数据的语言使用分析及应用

数学在一门学科中应用的程度，标志着它成熟的程度。

——恩格斯

8.1 概述

传统社会科学研究中的数据主要通过调查问卷或口头采访等方式获取，既耗时耗力，数据规模也很受限。进入互联网时代后，人类社会越来越多的信息以在线形式出现，为社会学研究提供了丰富的数据支持。特别是进入 Web 2.0 时代后，以用户为中心的服务（如微博、社交网站等）积累了大量的用户产生内容，包括用户个人档案（如性别、年龄、职业等信息）、用户社交关系网络（如关注关系、好友关系等）和文本信息（如微博、个人状态、博客等）等，成为社会学研究绝佳的数据来源。

顺应该趋势，2009 年由哈佛大学学者 David Lazer 牵头的来自信息科学、社会学和物理学的 15 位学者在 Science 杂志上联名发表文章，提出了“计算社会学”（Computational Social Science 或 Computational Sociology）（Lazer, et al. 2009），阐述了利用计算手段从大数据中揭示社会学规律的学术思想和趋势，标志着社会学进入到数据计算时代。短短几年内，计算社会学已成为人文社科领域近年来最重要的研究范式。Science、Nature 和美国国家科学院院刊等国际顶级学术期刊上大量涌现计算社会学的研究成果（Schich, et al. 2014, Lieberman, et al. 2007, Michel, et al. 2011, Bond, et al. 2012），众多学术期刊出版专刊介绍计算社会学研究进展。美国还成立了计算社会学学会，George Mason 大学甚至成立了计算社会学系，并成为世界上第一个正式授予计算社会学博士学位的单位。计算社会学无论对于揭示人类与社会规律，还是对于用户个性化服务，均具重要意义，因此基于社会媒体大数据的计算社会学研究，在学术界和产业界均引起广泛关注。

自然语言是社会媒体海量数据的重要组成部分，蕴藏了与用户及其复杂关系有关的丰富信息，是社会语言学、社会心理学等社会学分支的重要研究对象和研究角度，但是这些社会学分支所需的信息都隐藏在复杂的语言背后，需要利用自然语言处理和理解技术挖掘出来，才能被计算社会学研究进一步加以利用。随着机器学习和自然语言处理技术的发展，如何更好地分析社会媒体大数据中的自然语言已经成为计算社会学中的研究热点，近年来吸引了众多学者的研究兴趣，并已初具规模。

本文将综述最近在这方面的典型工作，并试图总结未来的研究趋势，希望对我国学术界和产业界在计算社会学的研发能够有所助益。

8.2 面向社会媒体的自然语言使用分析

传统的自然语言处理主要面向正式文本，例如新闻、论文等。这些文本遣词造句比较

规范，行文符合逻辑，因此比较容易处理。自然语言处理技术按照处理目标分为几个层次：（1）词汇层。主要是在词汇级别的处理任务，如中文分词、词性标注、命名实体识别等。（2）句法层。主要是在句法级别的处理任务，如针对句子的句法分析、依存分析等。（3）语义层。主要是在语义空间的处理任务，例如语义分析、语义消歧、复述等。（4）篇章层。主要是在篇章级别的处理任务，如指代消解、共指消解等。（5）应用层。主要是指利用自然语言处理分析技术完成的应用任务，如文本分类、信息抽取、问答系统、文档摘要、机器翻译，等等。关于自然语言处理技术的详细介绍可以参考（Jurafsky, et al. 2000，Mannin & Schütze 1999）。

进入社会媒体时代，用户产生的大量文本内容无论从词汇到造句都更加非正式，不仅存在大量拼写错误，还有很多网络产生的新用法，甚至出现专门的术语“网络用语”来命名这种现象。那么，自然语言处理技术如何分析社会媒体文本呢？研究者提出了文本正规化（text normalization）的任务，通过拼写纠错、词汇替换等方式，将非正式的网络文本转换为正式文本，然后再利用传统自然语言处理技术进行分析。当然这样还不够，研究者们还开始研究专门面向社会媒体文本特点的自然语言处理技术。

这里介绍的重点并不是面向社会媒体的自然语言处理技术，而是利用这些处理技术对社会媒体中的语言使用开展的分析工作。接下来，我们将介绍人们已经从社会媒体语言使用方面得到的主要成果。

8.2.1 词汇的时空传播与演化

词汇是自然语言的基本表意单位，也是自然语言处理的基础。利用词汇在时空中的变化开展社会学研究在国内外都不鲜见。 金观涛和刘青峰通过分析近代文献中的特定词汇使用情况，探讨了中国现代重要政治术语的形成（金观涛&刘青峰 2009）。最近，哈佛大学研究团队利用 Google Books 收集并扫描识别的 1800 年到 2000 年之间的 500 万种出版物（占人类所有出版物的 4%），通过不同关键词使用频度随时间的变化，分析了人类文化演进特点，做出了很多惊人的或有意思的发现。例如，他们发现在过去几百年里英语中越来越多的不规则变化动词演化成了规则变化动词（Lieberman, et al. 2007）。再如图 8.1 所示，通过 Google Books 中历年来使用“The United States is”和“The United States are”的统计趋势图，可以定量分析美国作为一个统一国家的概念是如何慢慢形成的（Aiden & Michel 2013）。他们甚至为此提出“文化组学”（culturomics，仿照“基因组学”发明的新术语）的概念（Aiden & Michel 2013，Michel, et al. 2011）。正如文献（Aiden & Michel 2013）的副标题“Big Data as a Lens on Human Culture”所暗示的，基于大数据的定量分析为社

会科学研究提供了一个全新的视角。

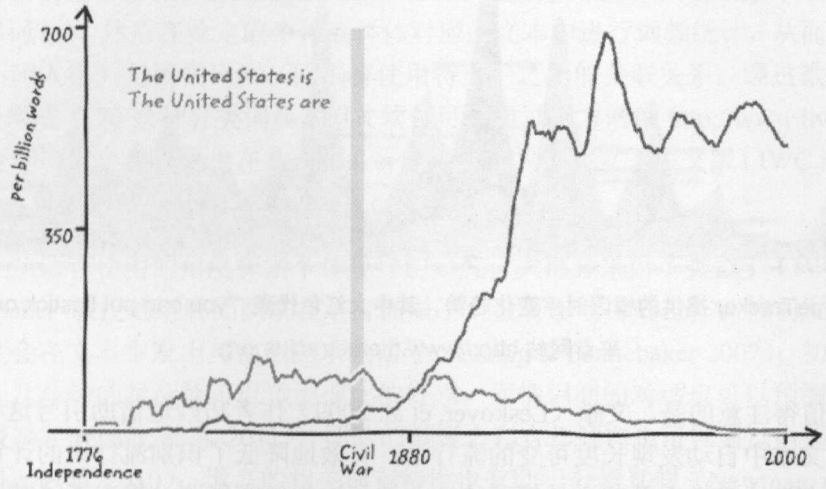


图 8.1 通过 Google Books 中历年来使用“The United States is”和“The United States are”的统计趋势图，可以定量分析美国作为一个统一国家的概念是如何慢慢形成的。来自文献（Aiden & Michel 2013）

在社会媒体中，新的词汇产生后，就会随着信息流动而进行传播和演化。一方面，新词汇的流行程度和形式会随着时间而演化，出现爆发（burst）和变形（variance）。不同新词汇的爆发程度和变形情况可能会受到不同因素的影响。另一方面，社会媒体中的用户分布在世界各地，其社交圈子往往会受到地理位置的限制，因此新词汇在社会媒体中用户间的传播，也会反映在地理位置的扩散上。一个词汇可能会首先在某个地域流行，然后逐渐扩散到全国，甚至全世界。

探索词汇的时空传播与演化，研究意义重大，相关技术也比较容易做到。目前已有关于英语词汇在社会媒体中的时空传播的研究。斯坦福大学 Leskovec 等人（Leskover, et al. 2009）从不同来源收集了约 9 千万篇新闻文章，利用引号从新闻中自动抽取流行语句，命名为模因（meme）。通过跟踪这些模因的使用频率随时间而变化的情况，能够及时、有效地把握美国政治、经济和文化生活，如图 8.2 所示。例如作者提到的典型模因“you can put lipstick on a pig”（为猪涂上口红）即是 2008 年美国总统大选中奥巴马讽刺竞选对手时引用的一句谚语，全句是“你就算给猪涂上口红，它也还是只猪”，当时引起了选民的广泛争议，也让最早出现于上世纪 20 年代的谚语“lipstick on a pig”重新流行起来，一时间成了美国人民很爱用的一个短语。通过文献（Leskover, et al. 2009），我们可以看到作者巧妙地使用了流行语作为社会热点问题的指标。

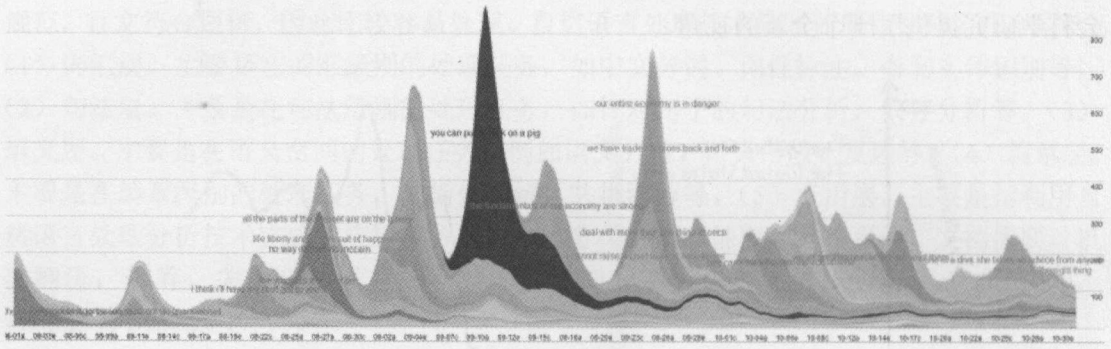


图 8.2 MemeTracker 提供的模因时序变化趋势，其中大红色代表“you can put lipstick on a pig”。来自网站 http://www.memetracker.org/

此外，值得注意的是，文献（Leskover, et al. 2009）作者巧妙地借助引号这种“显式标注”从海量文本中自动发现长度可变的流行语，有效地降低了识别流行语的计算难度。近年来，清华大学计算机系孙茂松教授系统地总结了这类研究思路，提出了“基于互联网自然标注资源的自然语言处理”的研究范式（孙茂松 2011），这对于如何有效利用大规模互联网数据具有极大的启发意义。Leskovec 研究团队还更进一步，通过聚类算法研究信息扩散的时序特征，分析 Twitter 和博客中模因使用的时序信息，共总结出 6 种时序曲线的主要形状（Yang & Leskover 2011）。

上述研究主要对流行语使用频率的时序变化进行了分析，也有学者考察了社会媒体中词汇与地域的关系。Eisenstein 等学者（Eisenstein, et al. 2010）发现同样的话题在不同地域会以不同的方式提出和讨论，为了探究 Twitter 中文本与使用者所处地域的关系，他们建立了一个瀑布模型（cascading model），用来分析词汇变化如何同时受到话题和地域的双重影响，并把地理空间按照语言学上的群体进行分割，试图通过文本本身去预测那些没有标注的用户所处的地域。词汇在地域上的差异和演化，与许多因素有关，如不同地域的文化风俗、地标建筑、方言俗语，等等。

词汇是文本中负载信息的基本单位，考察社会媒体中词汇的时空传播与演化，无论对语言演化研究，还是对社会管理，均具重要意义。

8.2.2 语言使用与个体差异

人格心理学和社会语言学的相关研究认为，人们的个体差异会反映在他们语言使用的特点上。因此，如何定量建立起语言使用与个体差异之间的关联，是学者关心的重要话题。这方面最具代表性的工作，是 20 世纪 90 年代 Pennebaker 和 King 提出的 Linguistic Inquiry

and Word Count（LIWC）方法（Pennebaker & King 1999）。其基本思想是以词汇作为语言使用定量分析的基本单位，首先通过人工收集、标注的方式建立不同类别的词典（如代词、数词、情感词等），然后在给定的个体或群体对应的文本中进行词频统计，从而建立起个体差异（即不同人格）与词类比例（即语言使用特点）之间的关联关系。经过数次修订后，LIWC 已经形成了 70 余种分类词典，相关软件可以通过官方网站 http://www.liwc.net/购买，而台湾地区学者黄金兰等人也在Pennebaker教授的授权下建立了中文版LIWC词典（Huang, et al. 2012），可以通过 http://cliwc.weebly.com/访问。

目前，从语言使用的角度探索个体差异的研究，大部分采用了类似于 LIWC 的研究范式。Pennebaker 教授的研究团队就在这方面做了大量有影响力的工作。他们发现，抑郁与自杀者往往会在文本中发出可侦测的求救信号（Chung & Pennebaker 2007）；初次约会的时候对象之间几分钟的对话就可以预测彼此的好感，而情侣间的对话也可以预测几个月后持续交往的概率（Ireland, et al. 2011）；团队的凝聚力和合作倾向也可以通过内部对话做出预测（Gonzales, et al. 2010）；谎言的相关语言特性也有助于分辨真假（Newman, et al. 2003）；语言使用分析还将有助于结识新朋友（Pennebaker & King 1999）；语言使用还与年龄有千丝万缕联系（Pennebaker & Stone 2003）等等。

然而，以上研究仍然未脱离传统社会学研究的藩篱，大部分是在受限的小规模数据上开展的。而在大规模在线社会媒体背景下，通过语言使用分析个体差异更凸显其重要性，一方面，很多在小规模数据上建立的社会理论需要在大规模真实数据进一步验证或再发现；而另一方面，利用社会媒体用户产生的文本数据推测用户的人格或心理特点，在个性化推荐服务中发挥重要作用。因此近年来，在社会计算领域提出了用户建档（user profiling）的研究任务，旨在利用用户产生内容预测用户的各种属性，既包括用户的各种简单属性，如性别（Burger, et al. 2011，Fink, et al. 2012）、年龄（Goswami, et al. 2009）和地理位置（Rao, et al. 2010，Li, et al. 2012）等，也包括用户的复杂属性，如兴趣（Yang, et al. 2011）、政治倾向（Rao, et al. 2010）、性格特点（Mairesse, et al. 2007，Schwartz, et al. 2013）和主观幸福感（Frank, et al. 2013，Mitchell, et al. 2013，Dodds, et al. 2011）等。

前述基于 LIWC 的研究与用户建档研究的主要不同在于：（1）前者侧重于人格差异与语言使用之间的关联关系的发现，而后者侧重于将语言使用作为特征来建立预测用户属性的模型。（2）前者更纯粹地考察语言使用与个体差异的关联，而后者则会将语言使用与用户的其他方面的特征（如用户的社会网络结构、在线行为模式等）综合起来进行属性预测。（3）前者对语言使用的分析还基本停留在词频统计的层面，而后者则充分利用了机器学习和自然语言处理领域的最新研究成果，如向量空间模型（Manning et al. 2008）、隐含主题模型（Steyvers & Griffiths 2007）、时间序列分析（Hamilton 1994）等，其定量分析的广度和

精度均为前者所不及。

目前面向大规模在线社会媒体的语言使用与个体差异的关系研究尚处于起步阶段，一方面在线社会媒体为研究提供了更丰富的分析素材和角度，而另一方面机器学习和自然语言处理的发展也为语言使用分析提供了更丰富的维度。可以预期，未来将能看到关于语言使用与个体差异的更多、更深层次的分析和发现。

8.2.3 语言使用与社会地位

语言是人类相互交流的工具，而社会中的人存在着地位差异。那么语言使用方式与人的地位差异有什么关系呢？这是一个社会语言学经典问题。

社会语言学理论提出，地位越低的发言者需要从语言上去适应地位越高的听者，而相反，地位越高的人则不需要调整自己的语言方式去适应别人（Gonzales, et al. 2010）。在过去由于缺少相关大规模数据，有关理论一直缺少定量分析的支持。美国康奈尔大学 Danescu-Niculescu-Mizil（以下简称 Mizil）等学者对这个问题进行了深入探讨，做出了一系列开创性的研究成果。

Mizil 等人（Danescu-Niculescu-Mizil, et al. 2012）选取线上和线下两个场景验证了交流行为是如何体现权力关系的。两个场景分别是维基百科中编辑的在线讨论，以及法庭庭审现场的辩护对话。值得注意的是，这里所谓的语言使用方式，并不是实词的使用，而是虚词的使用，甚至可能连发言者都没有注意自己这种发言方式的变化。该研究定量验证了参与讨论的人之间权力的差异会在两人如何回应对方的语言方式上有所体现。

该理论也在 Twitter 平台上得到了验证（Danescu-Niculescu-Mizil, et al. 2011）。首先，作者同样利用介词等虚词的使用情况，考察了交流双方的语言风格是如何彼此适应的。然后，作者考察了交流双方之间影响的不对称性，以及这种不对称性与社会地位的关系，即地位高的人不会去适应地位低的人，而地位低的人要付出更多去适应地位高的。研究结果表明，虽然 Twitter 对交流增加了一些限制（非面对面，非实时，而且只能说 140 个字），但交流中仍然有比较明显的语言适应行为。

礼貌用语的使用与社会地位之间也有密切关系（Danescu-Niculescu-Mizil, et al. 2013A）。作者分别对维基百科编辑和 Stack Exchange 论坛的讨论者进行研究，把用户对他人提出请求时的对话摘录出来，其中一句是真正的请求，而另一句是客套话，然后由标注者为其礼貌程度进行评价。研究结果表明，维基百科编辑在选举中试图获得更高地位时会更加礼貌，而一旦选上后，礼貌程度就会下降。这种情况也同样出现在 Stack Exchange 上，

人们的礼貌程度与地位呈反比关系。

该理论还被用来定量分析社区用户的语言使用变化情况（Danescu-Niculescu-Mizil, et al. 2013B）。作者以两个大型啤酒讨论社区作为研究对象，发现用户在社区中一般会经历两个阶段，在第一个阶段他们刚进入社区，会积极学习适应社区的语言使用规则，而接下来他们逐渐不再做出改变，任由规则变化，最后逐渐退出社区主流群体。该研究工作的学术意义在于，定量探索了在社区与个人的相互作用下，语言使用规则变化的复杂性。

Mizil 等人开创性地在社会媒体大数据上定量验证了社会语言学中的重要理论，并进一步利用该理论展开社会学研究。社会语言学乃至社会心理学中仍有大量的理论，有待于在大规模社会媒体中得到验证和利用，而语言使用是不可忽视的重要角度。

8.2.4 语言使用与群体分析

作为广大互联网用户在线交流信息和观点的平台，社会媒体汇集了成千上百万用户的产生内容，这些内容从整体上反映了人们关注的社会焦点和主要立场。从语言使用的角度，可以通过两个方面对这些用户进行群体分析：（1）作为文本内容的客观部分，分析用户群体关注的话题及其趋势；（2）作为文本内容的主观部分，分析用户群体的情绪、观点及其演化过程。

作为文本内容的客观部分，文本的话题检测与跟踪（Topic Detection and Tracking，简写为 TDT）（Allan 2002）是自然语言处理和信息检索领域的传统研究问题。最初是面向新闻媒体流提出的这个研究问题，旨在发现与跟踪新闻媒体流中的热点话题的趋势。在该任务中，一个话题是由一个种子事件及与其直接相关的事件组成的。在话题检测中有很多子任务，例如话题检测、话题跟踪、首次报道检测、关联检测，等等。面向社会媒体的话题检测与跟踪已经成为 TDT 的最新研究趋势，如图 8.3 是利用隐含主题模型分析 Twitter 话题并做可视化的样例，图 8.4 则是对 Twitter 话题变化趋势的分析与可视化。当然我们可以用单词或短语来表示话题，这样就可以利用 8.2.1 节“词汇的时空传播与演化”中的技术。但是，从实用角度，为了增强话题检测与跟踪的表达和概括能力，我们往往需要借助于隐含主题模型等技术，同时使用隐含主题和词汇一起来展示社会媒体的话题及其演化趋势，这是近年来的最新发展趋势。

图 8.3 利用主题模型分析 Twitter 话题并用标签云进行可视化。来自文献（Ramage，et al. 2010）

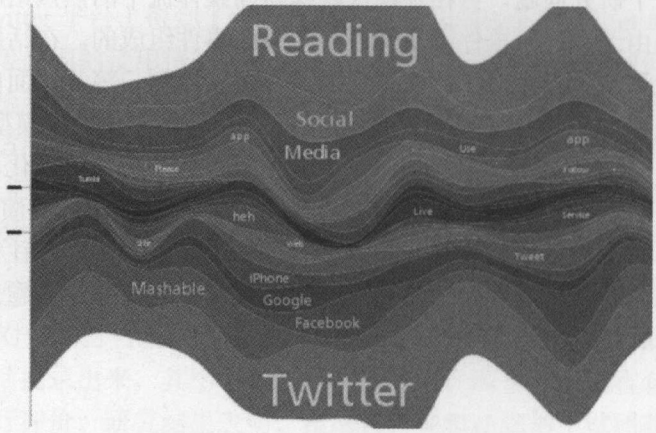


图 8.4 Twitter Stream Graphs 分析 Twitter 话题变化趋势并进行可视化

作为文本内容的主观部分，用户也会在社会媒体中表达他们的情绪、倾向和观点等主观情感。而社会媒体文本与传统媒体文本（如新闻）的最大不同也在于此，因此有大量研究聚焦于社会媒体的用户情绪和情感分析。如图 8.5 所示，作者通过分析 3 亿条 Twitter 数据中的情感词汇的使用情况，探索美国人的情绪随时间和地域的变化趋势，可以看到美国全国各地、一周七天以及每天 24 小时的情绪变化，得到很多有意思的结论。例如，美国人在下午的时候会变得烦躁，而在晚上开始好转；居住在美国西部的人普遍比东部沿海的人快乐，而位于美国南部的佛罗里达州几乎是最快乐的地方，等等。另外一个颇有影响力的工作是“We Feel Fine”项目，作者仅通过“We Feel X”的模板（其中 X 是待统计的情感词汇），在互联网博客等社会媒体中统计用户的情感分布，并用各种用户友好的可视化方案呈现给读者，可以很方便地查看不同类型用户（如男女、年龄）的主要情绪分布，如图 8.6 是该项目的搜索界面。可以说该工作也是充分利用互联网的海量、冗余的特点成功运用“基于互联网自然标注资源的自然语言处理”学术思想的典型代表。

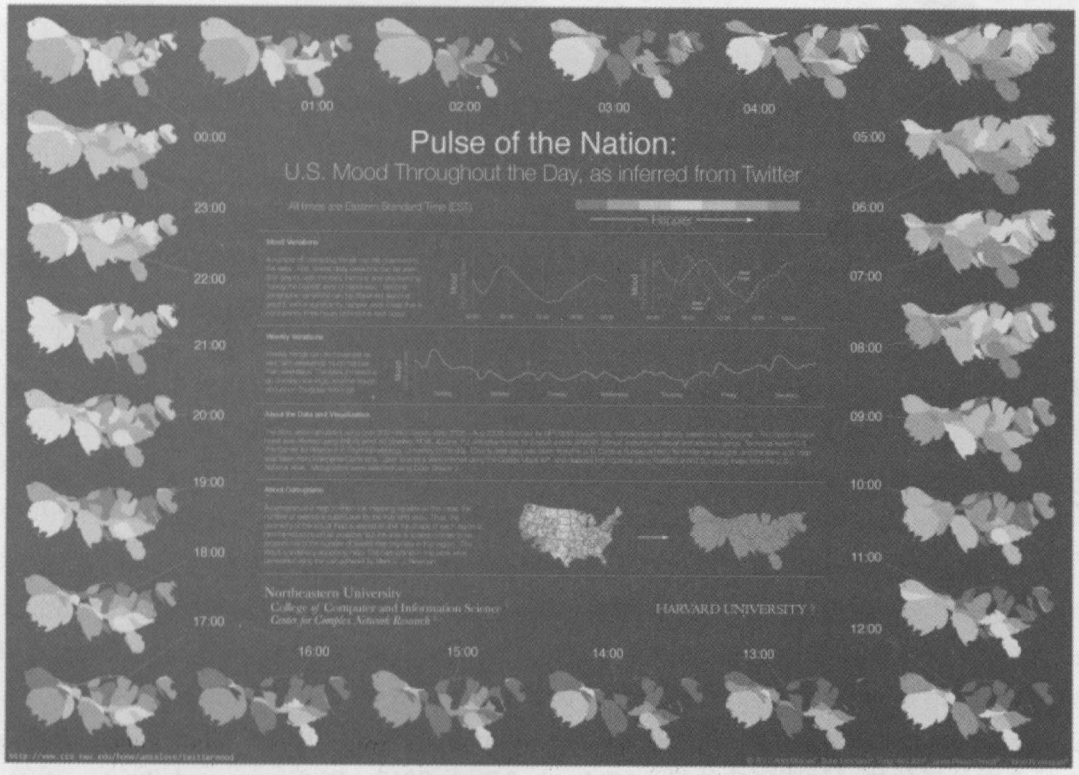


图 8.5 利用 Twitter 数据分析美国人情绪的时序变化。来自文献（Mislove, et al. 2010）

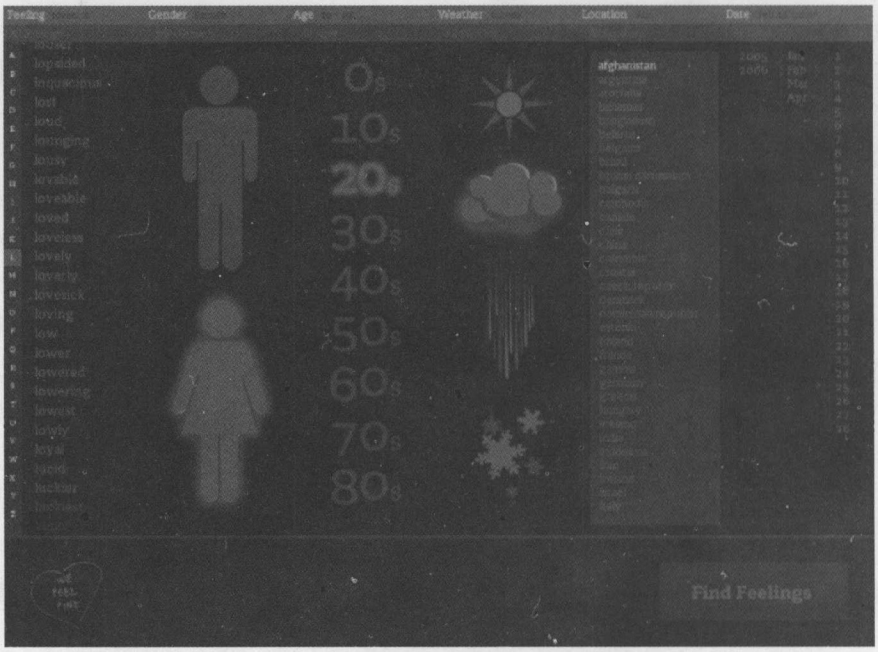


图 8.6 We Feel Fine 网站搜索界面。来自文献（Kamvar & Harris 2010）

8.3 面向社会媒体的自然语言分析应用

面向社会媒体中的自然语言分析技术有很多方面的应用，这里着重介绍几个有代表性的工作成果，相信在未来，会有更丰富而深入自然语言分析应用涌现出来。

8.3.1 社会预测

社会媒体用户产生内容在很大程度上反映了人们在社会生活方方面面的关注和立场，因此，最近被广泛用来进行各种社会事件的预测，包括产品销量（如电影票房收入）（Joshi, et al. 2010）、体育比赛结果（Sinha, et al. 2013）、股市走势（Bollen, et al. 2011，Zhang, et al. 2011）、政治选举结果（如美国总统大选）（Gross, et al. 2013，Yano, et al. 2013，Chung & Mustafaraj 2011，Williams & Gulati ，Tumasjan, et al. 2010，O'Connor, et al. 2010）、自然灾害传播趋势（如流行病传播）（St Louis &Zorlu 2012，Ritterman, et al. 2009），等等。

仅以政治选举为例，很多工作发现社会媒体中关于候选人的提及率就是很好的预测指

标，例如根据 Facebook 上的支持率就能够成功预测 2008 年美国总统大选结果（Williams & Gulati 2008）。更惊人的是，《信号与噪声》（Silver 2012）的作者 Nate Silver 在 2012 年准确预测了美国 50 个州的总统选举结果，虽然他不仅使用社会媒体中的信息，而是充分占据可获得的各类信息来进行预测，但是毫无疑问社会媒体在其中发挥了重要作用。2012 年 Nature 上发表的一篇题为《一个 6100 万人参与的关于社会影响和政治动员的实验》的文章（Bond, et al. 2012），则系统分析了 2010 年美国总统大选期间 Facebook 用户的相关情况，发现通过 Facebook 上的信息递送等社会动员（Social Mobilization），至少影响了现实世界中数以百万计人群的政治自我表达和投票行为。这说明，社会媒体不仅反映了人们的各种立场，可以用于预测，而且社会媒体还会对人们的现实生活产生深远的影响。在未来，如何将预测与干预有效结合，更好地分析、管理和利用社会媒体平台，将是身处于大数据中的每个政府、企业和政策制定者面临的重要课题。

毋庸置疑，由于社会媒体用户属性与现实社会的用户属性存在一定偏置，例如在我国，社会媒体上年轻人居多，收入相对较高，因此他们传达出来的关注与观点，并不能完全反映整个社会的立场和形势。因此，在近年来社会预测与干预研究轰轰烈烈开展的同时，也有人反思其有效性（Gayo-Avello 2012）。但纵观大势，随着移动设备的普及和互联网的发展，越来越多的人成为社会媒体用户，相信只要充分正视在线社会媒体与真实社会之间存在的偏差，我们就能够更好地利用社会媒体做好社会管理工作，更好地为人类生活服务。

8.3.2 霸凌现象定量分析

面向社会媒体的自然语言分析不仅可以用来进行社会预测，还可以用来支持解决社会公益问题，其中霸凌（bully）现象就是典型代表。霸凌是社会科学、尤其是青少年研究的经典研究课题。然而传统研究方法中这个课题的数据普遍量小、缺乏、对问题的呈现不够全面。而在社会媒体领域中关注这一话题的人士又普遍把视野局限在了网上欺负他人这个小范围内，没能够把线上线下的霸凌行为进行整合。最近有研究（Xu et al. 2012; Angela et al. 204）开始通过对 Twitter 上与霸凌有关的文本/叙述进行分析，而其关注的范围包括现实和虚拟环境中的欺负行为。

在这个研究中，先是从大量的 Twitter 博文中选取了与霸凌有关的作为原始数据，再主要进行四个方面的分析：文本分类（把含有霸凌关键词但并不相关的文本剔除）、角色判断（判断在欺负行为中是指责者、欺负者、受害者、报告者、还是其他）、感情分析、话题判断。该课题也证明，面向社会媒体的自然语言分析将有助于识别霸凌现象，及时干预，给予儿童更健康的生活环境。

8.4 未来研究的挑战与展望

关于面向社会媒体的自然语言分析及其应用，已然成为今年的研究热点，呈星火燎原之势，以上简介限于作者所见，难免有顾此失彼、挂一漏万之处，需要感兴趣的读者不断探索更多的研究成果和发现。然而，通过以上研究工作，我们可大致总结出面向社会媒体的自然语言分析及其应用的发展趋势。

（1）自然语言的深度分析。我们可以看到，仅基于词汇层（单词或短语）的简单统计，就已经产生了大量影响深远的研究工作。而近年来，伴随着互联网大数据爆发式增长，自然语言处理和机器学习领域飞速发展，未来将会有更多的自然语言深度分析的技术和工具不断成熟，例如自动根据大规模文档集合进行词汇语义聚类的隐含主题模型（Steyvers & Griffiths 2007），进行情感分析和观点挖掘的相关技术（Liu 2012）、进行跨语言分析的机器翻译技术（Koehn 2010）、对人类知识进行结构化管理和推理的知识图谱（Singhal 2012），等等。这些技术和工具的不断成熟和完善，将使我们面向社会媒体的分析如虎添翼，打开另一双天眼，可以看到以往所无法看到的世界，从而发现以往所不能发现的规律。

（2）跨媒体、跨平台、跨信息源的综合分析。从媒体类型而言，虽然社会媒体的出现对传统主流媒体（如国内各大新闻门户网站）产生重大冲击，但可以看到，主流媒体和社会媒体各有侧重、互为补充、深度交融，均为人们日常生活不可或缺的信息来源，很多情况下，主流媒体的相关新闻事件可以作为社会媒体分析的大背景，是分析人格特质的重要因素，例如探索人们在面临重大事件（如特大自然灾害）时的反应，等等；从社会媒体平台而言，无论是 Twitter 还是 Facebook，都只反映了人们生活的某个切面，例如以 Twitter 为代表的微博平台更具备自媒体特质，而以 Facebook 为代表的社会网络服务更具备好友圈特质，但这些平台背后都是同样的人，他们在不同平台上会有怎样不同的表现，以及这样的表现原因是什么，这既是社会学关心的话题，也是商业服务关心的问题，最近社会计算中的一个热门研究问题就是社会媒体跨平台的相同用户识别（Vosecky, et al. 2009，Liu, et al. 2013）；从信息源而言，社会媒体用户产生的内容非常丰富，包括文本、图像、社会网络以及大量结构化信息（如 Facebook 中的个人属性，虽然往往填写不完整、不准确），其中文本内容固然是重要组成部分，也是本书关注重点，但其他信息源亦扮演重要角色，例如大规模社会网络分析（Leskovec, et al. 2008）、大规模图像标注（Weston, et al. 2010），等等。未来，面向社会媒体的分析及其应用，需要将文本内容与其他信息源充分融合，进行跨媒体、跨平台的融合分析，只有充分进行跨媒体、跨平台和跨信息源的综合分析，才能发现人类社会更复杂、更深层的科学规律。

总之，面向社会媒体的自然语言分析与应用，无论对社会学和信息科学各领域的推进，

还是对商业服务的发展，均具有重要意义，日益引起人们的关注。其原因不言而喻，语言是人类区别于其他生物的最大特点，是进化厚赠人类的最珍贵礼物，也是人工智能、神经科学、社会语言学等领域孜孜以求希望真正理解的人类本质，还是人们进行日常交流、传承文化的重要载体。可以想象，随着社会媒体和互联网产生的海量数据，随着自然语言处理和机器学习等技术的高速发展，面向社会媒体的自然语言分析与应用必将大行其道，大有作为。

8.5 参考文献

[1] （Lazer, et al. 2009）Lazer D, Pentland A, Adamic L, et al. Computational Social Science[J]. Science. 2009, 323(5915): 721-723.

[2] （Schich, et al. 2014）Schich M, Song C, Ahn Y, et al. A network framework of cultural history[J]. science. 2014, 345(6196): 558-562.

[3] （Jurafsky, et al. 2000）Jurafsky D, Martin J H, Kehler A, et al. Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition[M]. MIT Press, 2000.

[4] (Mannin & Schütze 1999）Manning C D, H Schütze. Foundations of statistical natural language processing[M]. MIT Press, 1999.

[5] （金观涛 & 刘青峰 2009）金观涛，刘青峰. 观念史研究: 中国现代重要政治术语的形成[M]. 法律出版社, 2009.

[6] （Lieberman, et al. 2007）Lieberman E, Michel J, Jackson J, et al. Quantifying the evolutionary dynamics of language[J]. Nature. 2007, 449(7163): 713-716.

[7] （Aiden & Michel 2013）Aiden E, Michel J. Uncharted: Big Data as a Lens on Human Culture[M]. Penguin, 2013.

[8] （Michel, et al. 2011）Michel J, Shen Y K, Aiden A P, et al. Quantitative analysis of culture using millions of digitized books[J]. science. 2011, 331(6014): 176-182.

[9] （Leskover, et al. 2009）Leskovec J, Backstrom L, Kleinberg J. Meme-tracking and the dynamics of the news cycle[C]//Proceedings of WSDM. 2009.

[10] （孙茂松 2011）孙茂松. 基于互联网自然标注资源的自然语言处理[J]. 中文信息学报. 2011, 25(6): 26-32.

[11] （Yang & Leskover 2011）Yang J, Leskovec J. Patterns of temporal variation in online media[C]//Proceedings of WSDM. 2011.

[12] （Eisenstein, et al. 2010）Eisenstein J, O'Connor B, Smith N A, et al. A latent variable model for geographic lexical variation[C]//Proceedings of EMNLP. Association for Computational Linguistics, 2010.

[13] （Pennebaker & King 1999）Pennebaker J W, King L A. Linguistic styles: language use as an individual difference.[J]. Journal of personality and social psychology. 1999, 77(6): 1296.

[14] （Huang, et al. 2012）Huang C L, Chung C K, Hui N, et al. The development of the chinese linguistic inquiry and word count dictionary[J]. Chinese Journal of Psychology. 2012, 55(2): 185-201.

[15] （Chung & Pennebaker 2007）Chung C, Pennebaker J W. The psychological functions of function words[J]. Social communication. 2007: 343-359.

[16] （Ireland, et al. 2011）Ireland M E, Slatcher R B, Eastwick P W, et al. Language style matching predicts relationship initiation and stability[J]. Psychological Science. 2011, 22(1): 39-44.

[17] （Gonzales, et al. 2010）Gonzales A L, Hancock J T, Pennebaker J W. Language style matching as a predictor of social dynamics in small groups[J]. Communication Research. 2010, 37(1): 3-19.

[18] （Newman, et al. 2003）Newman M L, Pennebaker J W, Berry D S, et al. Lying words: Predicting deception from linguistic styles[J]. Personality and social psychology bulletin. 2003, 29(5): 665-675.

[19] （Pennebaker & Stone 2003）Pennebaker J W, Stone L D. Words of wisdom: language use over the life span.[J]. Journal of personality and social psychology. 2003, 85(2): 291.

[20] （Burger, et al. 2011）Burger J D, Henderson J, Kim G, et al. Discriminating Gender on Twitter[C]//Proceedings of EMNLP. Association for Computational Linguistics, 2011.

[21] （Fink, et al. 2012）Fink C, Kopecky J, Morawski M. Inferring Gender from The Content of Tweets: A Region Specific Example[C]//Proceedings of ICWSM. 2012.

[22] （Goswami, et al. 2009）Goswami S, Sarkar S, Rustagi M. Stylometric Analysis of Bloggers' Age and Gender[C]//Proceedings of ICWSM. 2009.

[23] （Rao, et al. 2010）Rao D, Yarowsky D, Shreevats A, et al. Classifying Latent User Attributes in Twitter[C]//Proceedings of The 2nd International Workshop on Search and Mining User-Generated Contents. 2010.

[24] （Li, et al. 2012）Li R, Wang S, Deng H, et al. Towards Social User Profiling: Unified and Discriminative Influence Model for Inferring Home Locations[C]//Proceedings of KDD. 2012.

[25] （Yang, et al. 2011）Yang S, Long B, Smola A, et al. Like Like Alike: Joint Friendship and Interest Propagation in Social Networks[C]//Proceedings of WWW. 2011.

[26] （Mairesse, et al. 2007）Mairesse F C C O, Walker M A, Mehl M R, et al. Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text[J]. Journal of Artificial Intelligence Research. 2007, 30: 457-500.

[27] （Schwartz, et al. 2013）Schwartz H A, Eichstaedt J C, Kern M L, et al. Personality, Gender, and Age in The Language of Social Media: The Open-Vocabulary Approach[J]. PLoS ONE. 2013, 8(9): e73791.

[28] （Frank, et al. 2013）Frank M R, Mitchell L, Dodds P S, et al. Happiness and the patterns of life: a study of geolocated tweets[J]. Scientific reports. 2013, 3.

[29] （Mitchell, et al. 2013）Mitchell L, Frank M R, Harris K D, et al. The geography of happiness: Connecting Twitter sentiment and expression, demographics, and objective characteristics of place[J]. PloS one. 2013, 8(5): e64417.

[30] (Dodds, et al. 2011) Dodds P S, Harris K D, Kloumann I M, et al. Temporal patterns of happiness and information in a global social network: Hedonometrics and Twitter[J]. PloS one. 2011, 6(12): e26752.

[31] （Manning et al. 2008）Manning C D, Raghavan P, Schütze H. Introduction to information retrieval[M]. Cambridge university press Cambridge, 2008.

[32] （Steyvers & Griffiths 2007）Steyvers M, Griffiths T. Probabilistic topic models[J]. Handbook of latent semantic analysis. 2007, 427(7): 424-440.

[33] （Hamilton 1994）Hamilton J D. Time series analysis[M]. Princeton university press Princeton, 1994.

[34] （Danescu-Niculescu-Mizil, et al. 2012）Danescu-Niculescu-Mizil C, Lee L, Pang B, et al. Echoes of power: Language effects and power differences in social interaction[C]//Proceedings of WWW. 2012.

[35] （Danescu-Niculescu-Mizil, et al. 2011）Danescu-Niculescu-Mizil C, Gamon M, Dumais S. Mark my words!: linguistic style accommodation in social media[C]//Proceedings of WWW. 2011.

[36] （Danescu-Niculescu-Mizil, et al. 2013A）Danescu-Niculescu-Mizil C, Sudhof M, Jurafsky D, et al. A computational approach to politeness with application to social factors[C]//Proceedings of ACL. Sofia, Bulgaria: Association for Computational Linguistics, 2013.

[37] （Danescu-Niculescu-Mizil, et al. 2013B）Danescu-Niculescu-Mizil C, West R, Jurafsky D, et al. No country for old members: user lifecycle and linguistic change in

online communities[C]//Proceedings of WWW. 2013.

[38] （Allan 2002）Allan J. Introduction to topic detection and tracking[M]. Topic detection and tracking, Springer, 2002, 1-16.

[39] （Ramage, et al. 2010）Ramage D, Dumais S T, Liebling D J. Characterizing Microblogs with Topic Models[C]//Proceedings of ICWSM. 2010.

[40] （Mislove, et al. 2010）Mislove A, Lehmann S, Ahn Y, et al. Pulse of the nation: Us mood throughout the day inferred from twitter[Z]. 2010.

[41] （Kamvar & Harris 2010）Kamvar S D, Harris J. We feel fine and searching the emotional web[C]//Proceedings of WSDM. 2011.

[42] （Joshi, et al. 2010）Joshi M, Das D, Gimpel K, et al. Movie reviews and revenues: An experiment in text regression[C]//Proceedings of HLT-NAACL. 2010.

[43] （Sinha, et al. 2013）Sinha S, Dyer C, Gimpel K, et al. Predicting the NFL using Twitter[J]. arXiv preprint arXiv:1310.6998. 2013.

[44] （Bollen, et al. 2011）Bollen J, Mao H, Zeng X. Twitter mood predicts the stock market[J]. Journal of Computational Science. 2011, 2(1): 1-8.

[45] （Zhang, et al. 2011）Zhang X, Fuehres H, Gloor P A. Predicting stock market indicators through twitter “I hope it is not as bad as I fear” [J]. Procedia-Social and Behavioral Sciences. 2011, 26: 55-62.

[46] （Gross, et al. 2013）Gross J, Acree B, Sim Y, et al. Testing the Etch-a-Sketch Hypothesis: A Computational Analysis of Mitt Romney's Ideological Makeover During the 2012 Primary vs. General Elections[C]//Proceedings of APSA. 2013.

[47] （Yano, et al. 2013）Yano T, Yogatama D, Smith N A. A penny for your tweets: Campaign contributions and Capitol Hill microblogs[C]//Media S I A C. 2013.

[48] （Chung & Mustafaraj 2011）Chung J E, Mustafaraj E. Can collective sentiment expressed on twitter predict political elections?[C]//Proceedings of AAAI. 2011.

[49] （Williams & Gulati 2008）Williams C, Gulati G. What is a social network worth? Facebook and vote share in the 2008 presidential primaries[C]//Proceedings of Annual Meeting of the American Political Science Association. 2008.

[50] （Tumasjan, et al. 2010）Tumasjan A, Sprenger T O, Sandner P G, et al. Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment[C]//Proceedings of ICWSM. 2010.

[51] （O'Connor, et al. 2010）O'Connor B, Balasubramanyan R, Routledge B R, et al. From tweets to polls: Linking text sentiment to public opinion time series[C]//Proceedings of ICWSM. 2010.

[52]（St Louis & Zorlu 2012）St Louis C, Zorlu G. Can Twitter predict disease outbreaks?[J]. BMJ. 2012, 344: e2353.

[53]（Ritterman, et al. 2009）Ritterman J, Osborne M, Klein E. Using prediction markets and Twitter to predict a swine flu pandemic[C]//Proceedings of International Workshop on Mining Social Media. 2009.

[54]（Silver 2012）Silver N. The signal and the noise: Why so many predictions fail-but some don't[M]. Penguin, 2012.

[55]（Bond, et al. 2012）Bond R M, Fariss C J, Jones J J, et al. A 61-million-person experiment in social influence and political mobilization[J]. Nature. 2012, 489(7415): 295-298.

[56]（Gayo-Avello 2012）Gayo-Avello D. " I Wanted to Predict Elections with Twitter and all I got was this Lousy Paper " --A Balanced Survey on Election Prediction using Twitter Data[J]. arXiv preprint arXiv:1204.6441. 2012.

[57]（Liu 2012）Liu B. Sentiment analysis and opinion mining[J]. Synthesis Lectures on Human Language Technologies. 2012, 5(1): 1-167.

[58]（Koehn 2010）Koehn P. Statistical machine translation[M]. Cambridge University Press, 2010.

[59]（Singhal 2012）Singhal A. Introducing the knowledge graph: things, not strings[J]. Official Google Blog, May. 2012.

[60]（Vosecky, et al. 2009）Vosecky J, Hong D, Shen V Y. User identification across multiple social networks[C]//Proceedings of International Conference on Networked Digital Technologies. 2009.

[61]（Liu, et al. 2013）Liu J, Zhang F, Song X, et al. What's in a name?: an unsupervised approach to link users across communities[C]//Proceedings of WSDM. 2013.

[62]（Leskovec, et al. 2008）Leskovec J, Backstrom L, Kumar R, et al. Microscopic evolution of social networks[C]//Proceedings of KDD. 2008.

[63]（Weston, et al. 2010）Weston J, Bengio S, Usunier N. Large scale image annotation: learning to rank with joint word-image embeddings[J]. Machine learning. 2010, 81(1): 21-35.

[64]（Xu et al, 2012）Jun-Ming Xu, Kwang-Sung Jun, Xiaojin Zhu, and Amy Bellmore. Learning from bullying traces in social media. In North American Chapter of the Association for Computational Linguistics - Human Language Technologies（NAACL HLT）. 2012.

[65]（Angela et al. 204）Angela J. Calvin, Amy Bellmore, Jun-Ming Xu, and Xiaojin Zhu. #bully: Uses of Hashtags in Posts about Bullying on Twitter. In Journal of School Violence, 2014.

后 记

大数据智能是一个快速发展的领域，当本书正式出版时，内容可能已经过时了。授人以鱼，不如授人以渔。因此，我想可以专门开辟一栏，介绍如何了解追踪大数据智能领域的最新学术资料。

国际学术组织、学术会议与学术论文

在计算机领域，国际上有众多专业学术组织，通过定期组织学术年会，报告学术论文，让学者们能够更方便地交流最新成果。这里以自然语言处理领域为例，介绍国际学术组织和学术会议的组织形式，以及国际学术论文的查找方式。

自然语言处理（NLP）在很大程度上与计算语言学（computational linguistics，CL）重叠。与其他计算机学科类似，NLP/CL 有一个属于自己的最权威的国际专业学会，叫 The Association for Computational Linguistics（ACL，URL：http://aclweb.org/ ），ACL 学会主办了 NLP/CL 领域最权威的国际会议，即 ACL 年会，ACL 学会还会在北美和欧洲召开分年会，分别称为 NAACL 和 EACL。除此之外，ACL 学会下设多个特殊兴趣小组（special interest groups，SIGs），聚集了 NLP/CL 不同子领域的学者，性质类似一个大学校园的兴趣社团。其中比较有名的诸如 SIGDAT（Linguistic data and corpus-based approaches to NLP）、SIGNLL（Natural Language Learning）等。这些 SIGs 也会召开一些国际学术会议，其中比较有名的就是 SIGDAT 组织的 EMNLP（Conference on Empirical Methods on Natural Language Processing）和 SIGNLL 组织的 CoNLL（Conference on Natural Language Learning）。此外还有一个 International Committee on Computational Linguistics 的老牌 NLP/CL 学术组织，它每两年组织一个称为 International Conference on Computational Linguistics（COLING）的国际会议，也是 NLP/CL 的重要学术会议。NLP/CL 的高水平学术论文主要分布在 ACL、NAACL、EMNLP 和 COLING 等几个学术会议上。

作为 NLP/CL 领域的学者最大的幸福在于，ACL 学会网站建立了称为 ACL Anthology 的页面（URL：http://aclweb.org/anthology-new/ ），支持该领域绝大部分国际学术会议论文的免费下载，甚至包含了其他组织主办的学术会议，例如 COLING、IJCNLP 等，并支持基于 Google 的全文检索功能，可谓一站在手，NLP 论文我有。由于这个论文集合非常庞大，并且可以开放获取，很多学者也基于它开展研究，提供了更丰富的检索支持，具体入口可以参考 ACL Anthology 页面上方搜索框右侧的不同检索按钮。

与大部分计算机学科类似，由于技术发展迅速，NLP/CL 领域更重视发表学术会议论文，原因是发表周期短，并可以通过会议进行交流。当然 NLP/CL 也有自己的旗舰学术期刊，发表过很多经典学术论文，那就是 Computational Linguistics（URL：http://www.mitpressjournals.org/loi/coli）。该期刊每期只有几篇文章，平均质量高于会议论文，时间允许的话值得及时追踪。此外，ACL 学会为了提高学术影响力，也刚刚创办了 Transactions of ACL（TACL，URL：http://www.transacl.org/ ），值得关注。值得一提的是这两份期刊也都是开放获取的。此外也有一些与 NLP/CL 有关的期刊，如 ACM Transactions on Speech and Language Processing，ACM Transactions on Asian Language Information Processing，Journal of Quantitative Linguistics 等等。

根据 Google Scholar Metrics 2015 年对 NLP/CL 学术期刊和会议的评价，ACL、EMNLP、NAACL、COLING、LREC、Computational Linguistics 位于前 5 位，基本反映了本领域学者的关注程度。

值得一提的是，美国 Hal Daumé III 维护了一个 natural language processing 的博客（http://nlpers.blogspot.com/ ），经常评论最新学术动态，值得关注。我经常看他关于 ACL、NAACL 等学术会议的参会感想和对论文的点评，很有启发。另外，ACL 学会维护了一个 Wiki 页面（http://aclweb.org/aclwiki/ ），包含了大量 NLP/CL 的相关信息，如著名研究机构、历届会议录用率，等等，都是居家必备之良品，值得深挖。

NLP/CL 是文本大数据智能的主要研究领域。作为交叉学科，大数据智能也与以下几个方面密切相关：（1）信息检索和数据挖掘领域。相关学术会议主要由美国计算机学会（ACM）主办，包括 SIGIR、WWW、WSDM、KDD 等。相关期刊包括 ACM TOIS、ACM TKDD、IEEE TKDE 等。（2）人工智能领域。相关学术会议主要包括 IJCAI 和 AAAI 等，相关学术期刊主要包括 Artificial Intelligence 和 Journal of AI Research。（3）机器学习领域，相关学术会议主要包括 ICML，NIPS，AISTATS，UAI 等，相关学术期刊主要包括 Journal of Machine Learning Research（JMLR）和 Machine Learning（ML）等。

中国计算机学会（CCF）制定了“中国计算机学会推荐国际学术会议和期刊目录”

（http://www.ccf.org.cn/sites/ccf/aboutpm.jsp?contentId=2567814757463），基本公允地列出了每个领域的高水平期刊与会议。大家可以通过这个列表，迅速了解每个领域的主要期刊与学术会议。

值得注意的是，虽然计算机领域学术会议论文的发表周期已经非常短，仍然不能满足最近深度学习等方向的迅猛发展。因此，越来越多学者选择绕过学术会议或期刊的审稿流程，直接通过 arXiv（http://arxiv.org/）等预印本平台在线发布论文。由于省去了同行评议的流程，这些最新学术成果得以更快地发布。但也由于缺少同行评议的意见和过滤，导致预印本平台上发布的论文质量良莠不齐，需要有较强的鉴别力，才能找到其中真正有价值的工作。现在，虽然自然语言处理等方向的学者对通过 arXiv 率先发布成果看法不一，众说纷纭，但毋庸置疑 arXiv 已经成为深度学习最新进展的重要发布渠道，Yoshua Bengio 等著名学者及其团队的最新研究成果，往往先发布在 arXiv 上，然后再发表在相关顶级会议上。因此，arXiv 是了解大数据智能最新进展的重要信息渠道。

国内学术组织、学术会议与学术论文

与国际上相似，国内也有一家与 NLP/CL 相关的学术组织，中国中文信息学会（URL：http://www.cipsc.org.cn/ ）。通过学会的理事名单（http://www.cipsc.org.cn/lingdao.php ）基本可以了解国内从事 NLP/CL 的主要单位和学者。中文信息学会每年组织很多学术会议，例如全国计算语言学学术会议（CCL）、中国自然语言处理青年学者研讨会（YSSNLP）、全国信息检索学术会议（CCIR）、全国机器翻译研讨会（CWMT）等，是国内 NLP/CL 学者进行学术交流的重要平台。尤其值得一提的是，中国自然语言处理青年学者研讨会是专门面向国内 NLP/CL 青年学者的研讨交流会，采用邀请制参加，大家自愿报名在研讨会上报告学术前沿动态，是国内 NLP/CL 青年学者进行学术交流、建立学术合作的绝佳平台。

2010 年的 COLING 和 2015 年的 ACL 在北京召开，均由中文信息学会负责组织工作，这在一定程度上反映了学会在国内 NLP/CL 领域的重要地位。此外，计算机学会中文信息技术专委会组织的自然语言处理与中文计算会议（NLP&CC）是最近崛起的国内重要 NLP/CL 学术会议。中文信息学会主编了一份历史悠久的《中文信息学报》，是国内该领域的重要学术期刊，发表过很多篇重量级论文。此外，国内著名的《计算机学报》、《软件学报》等期刊上也经常有 NLP/CL 论文发表，值得关注。

过去几年，在水木社区 BBS 上开设的 AI、NLP 版面曾经是国内 NLP/CL 领域在线交流讨论的重要平台。这几年随着社会媒体的发展，越来越多学者转战新浪微博，有浓厚的

交流氛围。如何找到这些学者呢？一个简单的方法就是在新浪微博搜索的“找人”功能中检索“自然语言处理”、“计算语言学”、“信息检索”、“机器学习”等字样，马上就能跟过去只在论文中看到名字的老师同学们近距离交流了。还有一种办法，清华大学梁斌开发的“微博寻人”系统（http://xunren.thuir.org/）可以检索每个领域的有影响力人士，因此也可以用来寻找 NLP/CL 领域的重要学者。值得一提的是，很多在国外任教的老师和求学的同学也活跃在新浪微博上，例如王威廉（http://weibo.com/u/1657470871）、李沐（http://weibo.com/mli65）等，经常发布重要的业内新闻，值得关注。微博上还有一些专门整理和推送人工智能相关信息的账号，如好东西传送门（http://memect.com/），每天都会整理发布的每日领域动态。学术研究既需要苦练内功，也需要与人交流。所谓言者无意、听者有心，也许其他人的一句话就能点醒你苦思良久的问题。毫无疑问，微博等社交媒体提供了很好的交流平台，但也要注意不宜沉迷。

如何快速了解某个领域的研究进展

最后简单说一下快速了解某领域研究进展的经验。你会发现，搜索引擎是查阅文献的重要工具，尤其是谷歌提供的 Google Scholar，由于其庞大的索引量，将是我们披荆斩棘的利器。

当需要了解某个领域，如果能找到一篇该领域的最新研究综述，就省劲多了。最方便的方法还是在 Google Scholar 中搜索“领域名称 + survey / review / tutorial / 综述”来查找。也有一些出版社专门出版各领域的综述文章，例如 NOW Publisher 出版的 Foundations and Trends 系列，Morgan & Claypool Publisher 出版的 Synthesis Lectures on Human Language Technologies 系列等。它们发表了很多热门方向的综述，如文档摘要、情感分析和意见挖掘、学习排序、语言模型等。

如果方向太新还没有相关综述，一般还可以查找该方向发表的最新论文，阅读它们的“相关工作”章节，顺着列出的参考文献，就基本能够了解相关研究脉络了。当然，还有很多其他办法，例如去 videolectures.net 上看著名学者在各大学术会议或暑期学校上做的 tutorial 报告，去直接咨询这个领域的研究者，等等。

十载耕耘奠定专业地位

博文视点诚邀精锐作者加盟

以书为证彰显卓越品质

《C++Primer（中文版）（第5版）》、《淘宝技术这十年》、《代码大全》、《Windows内核情景分析》、《加密与解密》、《编程之美》、《VC++深入详解》、《SEO实战密码》、《PPT演义》……

“圣经”级图书光耀夺目，被无数读者朋友奉为案头手册传世经典。

潘爱民、毛德操、张亚勤、张宏江、咎辉Zac、李刚、曹江华……

“明星”级作者济济一堂，他们的名字熠熠生辉，与IT业的蓬勃发展紧密相连。

十年的开拓、探索和励精图治，成就**博**古通今、**文**圆质方、**视**角独特、**点**石成金之计算机图书的风向标杆：博文视点。

“凤翱翔于千仞兮，非梧不栖”，博文视点欢迎更多才华横溢、锐意创新的作者朋友加盟，与大师并列于IT专业出版之巅。

英雄帖

江湖风云起，代有才人出。
IT界群雄并起，逐鹿中原。
博文视点诚邀天下技术英豪加入，
指点江山，激扬文字
传播信息技术，分享IT心得

• 专业的作者服务 •

博文视点自成立以来一直专注于IT专业技术图书的出版，拥有丰富的与技术图书作者合作的经验，并参照IT技术图书的特点，打造了一支高效运转、富有服务意识的编辑出版团队。我们始终坚持：

善待作者——我们会把出版流程整理得清晰简明，为作者提供优厚的稿酬服务，解除作者的顾虑，安心写作，展现出最好的作品。

尊重作者——我们尊重每一位作者的技术实力和生活习惯，并会参照作者实际的工作、生活节奏，量身制定写作计划，确保合作顺利进行。

提升作者——我们打造精品图书，更要打造知名作者。博文视点致力于通过图书提升作者的个人品牌和技术影响力，为作者的事业开拓带来更多的机会。

联系我们

博文视点官网：http://www.broadview.com.cn　　CSDN官方博客：http://blog.csdn.net/broadview2006/

投稿电话：010-51260888　88254368　　投稿邮箱：jsj@phei.com.cn

新浪微博 weibo.com　@博文视点Broadview　　微信公众账号　博文视点Broadview

博文视点精品图书展台

专业典藏

移动开发

大数据 · 云计算 · 物联网

数据库

Web开发

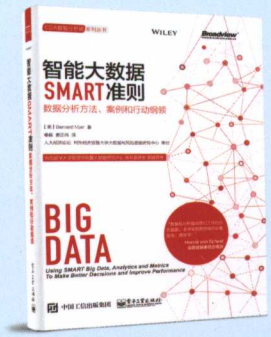
程序设计

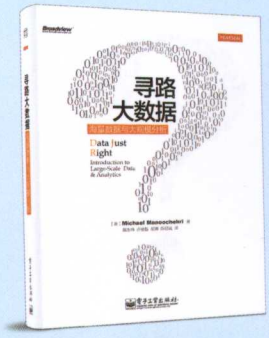
软件工程

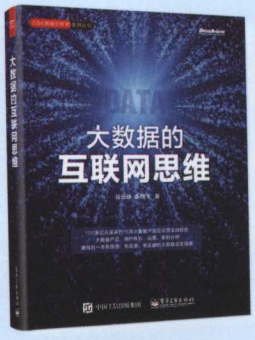
办公精品

网络营销

好书分享







大数据文摘

专注大数据，每日有分享

数据派

清华大数据产业联合会官方微信

Broadview®
www.broadview.com.cn
博文视点·IT出版旗舰品牌
技术凝聚实力·专业创新出版

大数据智能

互联网时代的机器学习和自然语言处理技术

本书是一本介绍大数据智能分析的科普书籍，旨在让更多的人了解和学习互联网时代的机器学习和自然语言处理技术，以期让大数据技术更好地为我们的生产和生活服务。

全书包括大数据智能基础和大数据智能应用两个部分，共8章。大数据智能基础部分有三章：第1章以深度学习为例介绍大数据智能的计算框架；第2章以知识图谱为例介绍大数据智能的知识库；第3章介绍大数据背后的计算处理系统。大数据智能应用部分有五章：第4章介绍智能问答，第5章介绍主题模型，第6章介绍个性化推荐系统，第7章介绍情感分析与意见挖掘，第8章介绍面向社会媒体大数据的语言使用分析及应用。最后在本书的后记部分为读者追踪大数据智能的最新学术材料提供了建议。

博文视点Broadview

新浪微博 weibo.com
@博文视点Broadview

统筹策划：顾慧芳
责任编辑：徐津平
封面设计：赵存存

上架建议：大数据

ISBN 978-7-121-27648-4

定价：49.00元